Group name/number

Group 1 – ***Team Jupiter***

Project plan

Our project name is "**Patient Management System**" where the main target of this project is to manage the patient in a hospital with a proper and well-constructed way so that the spending time complexity of each and every patient will be reduced. For this reason, the doctors as well as the other medical co-operative will be able to give their services to the large amount of people by spending less time table.

Group Member

Name	Student ID	Section
Md Ashif Ur Rahman	171-15-1162	PC-B
Md Masud Rana	171-15-1370	
Md Monir Hossain Shuvo	171-15-1340	
Naimur Rahman Nafio	171-15-1196	
Md Nahid Nibir	173-15-1692	

Version history

Version	Date	Author	Description
0.0	**23.10.2020**	**Md Ashif Ur Rahman**	**First draft**
0.1			

Contents

Introduction

Purpose and scope of project

"Patient Management System" is a development project funded by the Ministry of Health & Family Welfare for developing an efficient and well updated with less time-consuming patient management system for the government hospital and medical college hospital. One of the subprojects of "Patient Management System" focuses on developing a national Patient handling app. "**RogirSeba**" is a web-based app where the doctors, Pathologists, Pharmacist can take care and observe patient's present situation with close eyes. The students have both the role of the student and of the teacher. The main features of this web-based app are that doctors with other hospital co-operatives can update the present situation, view and cancel the pathological reports, family history reports, other surgical with other disease symptom reports, medicine, upload a profile photo of patients, no invitation limit, and so on. The aim is to create a friendly, unintimidating platform where both patients and doctors feel comfortable to delivery medical service to the patients.

Purpose of this document, scope of the project.

- To give medical services to the patients more efficiently.
- To save documents, reports and tests of a particular patients for further medical checkup and services.
- To remember the consultants about the treatment of a particular patients for furter and updated treatments.

Goal of this document is to help groups to start their project successfully ("not forgetting anything"), and manage their processes and projects.

- To give dynamic medical services to the patients.
- To reduce the working time of medical consultants of patients to use it more effectively.

Product and environment

Many higher education institutions in Finland have had their own tandem courses where the selection of languages has often been restricted. This means that not every student has been able to participate in the course and not able to learn a language they would like to. In UniTandem, however, the selection of languages and their combinations are not restricted. Since it is an online-based application, students can find a language partner from any of the institutions registered to the app.

Intended User Groups

Product name, high-level business goals and the environment in general. You may add some kind of concept picture/diagram about this software and its environment and possible other related/depending systems.

What is/are the intended user group(s) ? E.g. students, engineers, shop assistants,…

- Doctors
- Patients
- Pathologists
- Pharmacologists
- Medical representatives

Customer's current system and other similar systems

Symplex himes, AtenaOne, EagleSoft etc are a great starting point for beginners to build vocabulary and make first steps. Most of the applications are subscription-based, small language options, unorganized, and so on.

How customer does the work now, or is this a completely new system?

Are there any existing similar or look-alike systems? Feel free to “spy” already existing systems from public sources (e.g. make a quick info search on web, as is done in all smart projects).

Project constraints

Existing code branch that we have to extend

Technology selection freedom is not there

Time limitations, with other courses and so on.

Existing production system is there, so we need to ensure this system update won't harm production data.

Such as technology constraints, development constraints, customer's guidelines, coding conventions, directives, laws, and standards. All which are already known.

Practices, methods and tools set by the customer (usually according their existing processes, e.g. follow company GUI design).

Licensing issues, e.g. use of 3rd party components.

IPRs (if already agreed in some way, by default students').

Definitions, abbreviations and acronyms

For example:

basic user university course assistant

end-user first year student

PB Product Backlog

SB Sprint Backlog

super-user admin user

UI User interface.

(HINT: if you use the word "normal" user, what is then "abnormal user" ?)

Customer
Related organizations

Requirements

Functional requirements (main goals)

Main functional requirements are listed below. This mainly includes some fixes in the existing system. Backlog should have them in detail.

Admin:

- Be able to upload profile photo more than 3.5mb
- News on the main page should be shown latest to oldest. (currently this is working opposite)
- Sign-ups for few email domains are rejected. This should work based on given whitelisted domains.

General User:

- Update links to right ones for Digi campus Moodle. Links will be given.
- Sent requests should be visible somewhere and ability to cancel them.
- Regardless of receiver or sender, 10 requests are limited now. This needs to be changed according to client's requirement

IT Officer:

- Report on the applications security. Possibly improve.
- Report on the error handling, and logs. Possibly improve.
- Implement HAKA login. (Least priority)

Non-functional requirements goals

Usability goals

The usability goal is to have an application which doesn't have usability issues. The goal also is that everything in the user interface is logical and clear to use. In addition the user interface is pretty simple and boring, so something could be done to the user interface.

The application front page needs changes.

We need to have a usability test with the application.

Performance goals
Some mechanism may use some refactor which might lead to a shorter time of execution and increase performance.

Reliability goals
None, at the moment.

Security goals
As the main functionality has a goal related to this, performing that security analysis of the application there might bring up something.

User interface requirements (main goals)

If user interface design is critical for your system, briefly sketch the main views/states here. Use example pictures of views, menus and dialogs, if possible and appropriate. Use separate sections for different views.

You may also write a separate GUI design plan, if so put reference here.

(HINT: remember to ask end-users' opinion, too.)

Risk management

Risk management is the process of identifying, monitoring, assessing and neutralizing threats and roadblocks to a project. Risks could stem from a wide variety of situations such as technical complexities, dropouts, management errors etc. Severe scenarios may cause the project to fail partially or completely.

Implementing a risk management plan could significantly help reduce the chances of a project failure. Carefully monitoring and estimation of potential risks shall certainly help making the project a success.

Risk list

List the identified risks or summary at the start of the project.

A separate and updated risk list is maintained at MMT.

Table 7.1. Project risks.

Risk ID	Explanation, severity/impact, probability, size/importance
Dropouts	**Chances are that members might dropout from the project due to motivation or complexity of code/concept.** **Impact: Low** **Probability: Low** **Importance: This might lead to increase in other members tasks and as a result this could happen for them.**
Estimation and scheduling	**Some tasks may be underestimated/overestimated and the scheduled wrong. This may increase time to reach the goals.** **Impact: Medium** **Probability: Medium** **Importance: To ensure project deadline is utilized well, this needs to be done carefully.**
Productivity issues	**Productivity might be low for some because of unfamiliarity of the technology used.** **Impact: High** **Probability: Medium** **Importance: This might have a high impact since getting tasks done may include longer time than estimated.**

Gold plating	**Premature optimization or doing unnecessary implementation costs the project time.** **Impact: High** **Probability: Medium** **Importance: This may happen passively, without the developer noticing. Proper code reviews should be done in order to avoid it.**
Technical risk	**Some difficulties might come up setting up or adding new tools/technology to the existing solution to achieve a task. That may prove challenging.** **Impact: High** **Probability: Low** **Importance: Careful planning should help prevent this.**
Inadequate resource	**The project size might be quite big for the available resources considering not everyone has deep understandings of the technologies used.** **Impact: Medium** **Probability: High** **Importance: This may cause a delay on getting started with the tasks.**

Risk monitoring

Monitoring risks and keeping them in checks are essential to avoid any unexpected circumstances that may jeopardize the project.

Risk management is done at MMT tool.

Steps we would take to monitor risks are:

- Weekly meetings
- Task progress
- Re-assign task to another person if necessary

References

[1] W3Schools. Node.js Tutorial. Available at: https://www.w3schools.com/nodejs/. Last read 15/9/2020.

[2]

Open issues

……..

Ideas for further development

1. Adding the possibility to install the application or add it to desktop

2. Improving the offline user experience (at the moment, when offline, the standard dinosaur game of Chrome begins)

3. Suggesting a UI revamp

APPENDIX A […Z]

The original task list provided by the client