Real-Time Demand Capacity Tool

Project Definition

Olivier Clermont (6445938)  
Jonathan Ermel (6408238)  
Mathieu Fortin-Boulay (6571541)  
Philippe Legault (6376254)  
Nicolas Ménard (6357275)

Presented to Prof. Liam Peyton  
for the Software Engineering Capstone Project   
SEG4910

University of Ottawa  
January 23, 2015

Table of Contents

[Project Title & Team 3](#_Toc409102337)

[Customer 3](#_Toc409102338)

[Objectives 3](#_Toc409102339)

[System 3](#_Toc409102340)

[Architecture Description 10](#_Toc409102341)

[Risk Assessment 10](#_Toc409102342)

[Engineering Challenges 10](#_Toc409102343)

[Organizational Challenges 11](#_Toc409102344)

[Deployment Challenges 11](#_Toc409102345)

[Impact Assessment 11](#_Toc409102346)

[Legal Issues 11](#_Toc409102347)

[Liability Issues 11](#_Toc409102348)

[Societal Issues 12](#_Toc409102349)

[User Community 12](#_Toc409102350)

[Financial Impact 12](#_Toc409102351)

[Project Plan 12](#_Toc409102352)

[Iteration 1 - Project Definition 12](#_Toc409102353)

[Iteration 2 - Requirements Analysis 13](#_Toc409102354)

[Iteration 3 - Demo & Quality Assurance Plan 13](#_Toc409102355)

[Iteration 4 - Alpha & Architecture Design Report 13](#_Toc409102356)

[Iteration 5 - Beta & Quality Assurance Report 13](#_Toc409102357)

[Iteration 6 - Deployed & Final Report 13](#_Toc409102358)

# Project Title & Team

The application being developed will be referred to as **RTDC** (Real-Time Demand Capacity). Listed below are the team members collaborating on the project:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Student Number | Email | Co-op Status |
| Olivier Clermont | 6445938 | ocler043@uottawa.ca | Non-co-op |
| Jonathan Ermel | 6408238 | jerme043@uottawa.ca | Non-co-op |
| Mathieu Fortin-Boulay | 6571541 | mfort059@uottawa.ca | Co-op |
| Philippe Legault | 6376254 | plega060@uottawa.ca | Co-op |
| Nicolas Ménard | 6357275 | nmena080@uottawa.ca | Co-op |

# Customer

The system will be developed for Health Now. It will be used at the Montfort Hospital and the Queensway Carleton Hospital as part of a pilot study. Being in the healthcare business, it is their mission to provide excellent care and support to every patient. All this must be done in a timely fashion in order to encourage quick recoveries all while cutting down on operation costs.

Throughout the project, we will be interacting with Mr. Alain Mouttham (amouttham@rogers.com), an employee of the Queensway Carleton Hospital. Mr. Mouttham holds a Master’s in Computer Science and is co-founder of startups based in Silicon Valley and Ottawa. He also co-authored several healthcare studies and is now focusing on improving Patient Flow Management.

# Objectives

Bed overflow is an increasing problem in hospitals across Canada. However, studies suggest that patient flow can be improved by assessing capacity and demand in real-time. Therefore, our team has been given the task of developing a cross-platform tool that helps hospital staff implement the Real-Time Demand Capacity (RTDC) method. The application will help staff monitor bed availability and execute plans to free up beds. In addition to presenting real-time information concerning demand and capacity, the tool will provide unified communications. Once integrated with their system, the Queensway Carleton Hospital will be able to judge the efficiency of the RTDC method. The project shall be deemed successful if it allows users to assess capacity, estimate demand, build and implement plans, evaluate plan execution, and communicate between each other.

# System

There are two main actors in the application. The first is a manager who is responsible for overseeing the capacity and demand of each hospital unit. It is the user that creates action plans, each of which contains a set tasks to execute in order for patients to be discharged.

The second actor is an employee. Employees work toward completing tasks which were assigned to them or their group by a manager. An employee may require the assistance of another employee in order to complete his/her task. Therefore, the application will provide employees with a method of communicating between each other via messages, voice calls and video calls.

A use case diagram illustrating the actors and the system’s main functionality is included on the next page.



Figure - Use case diagram

The following pages include a mockup of the user interface.

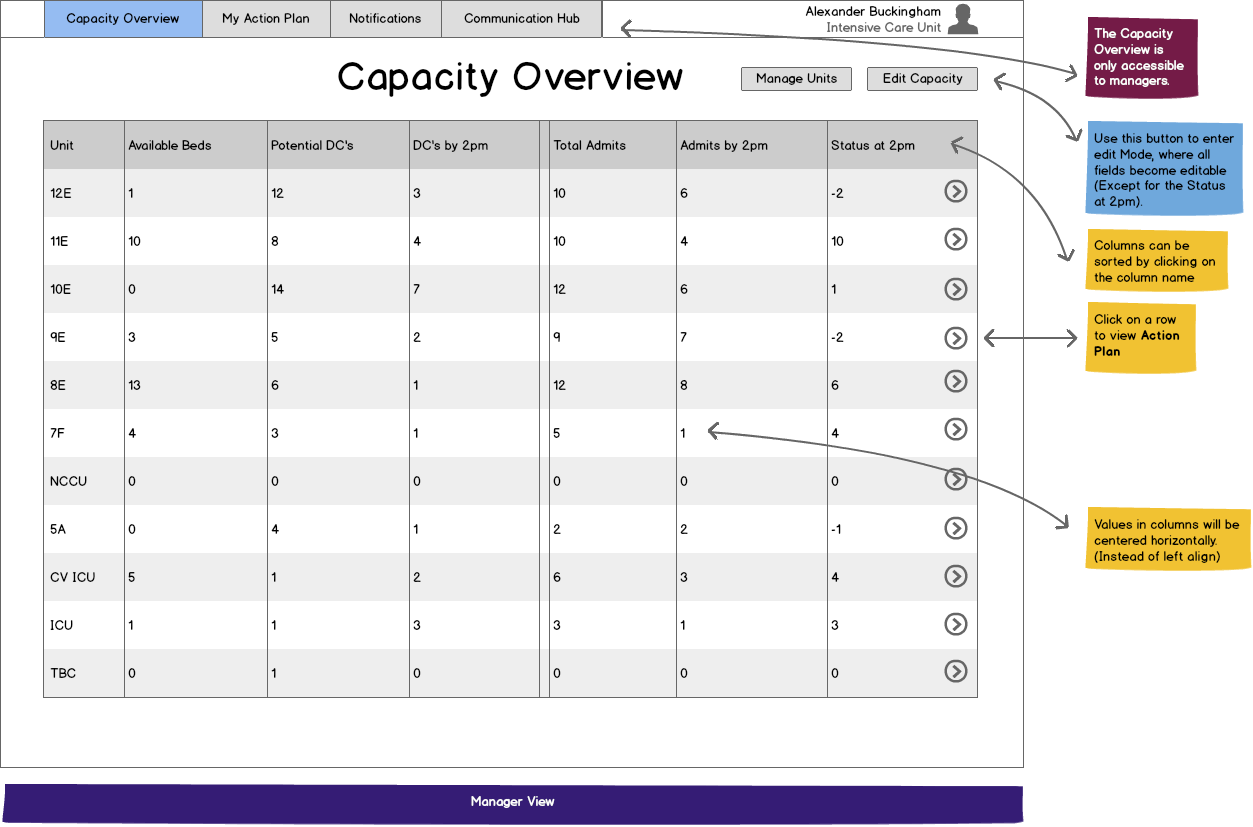


Figure - Capacity Overview

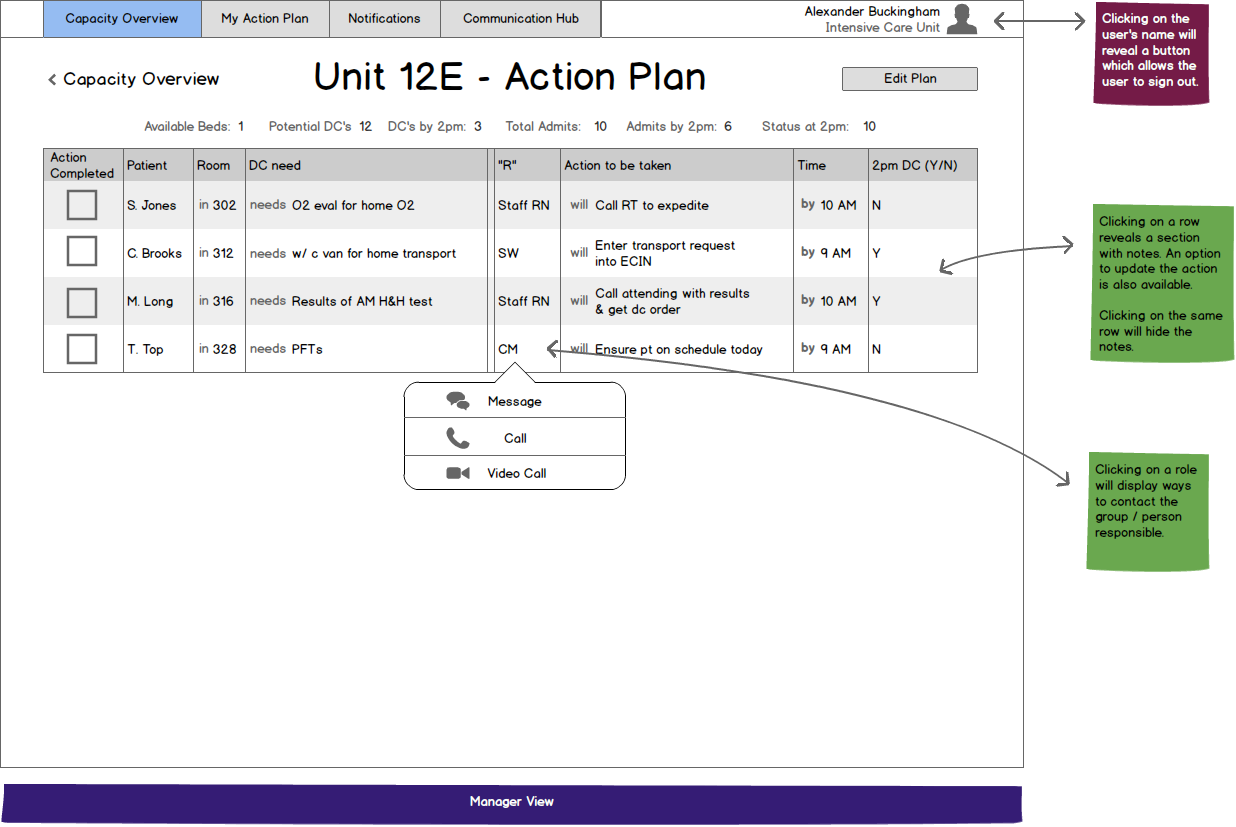


Figure - Unit action plan

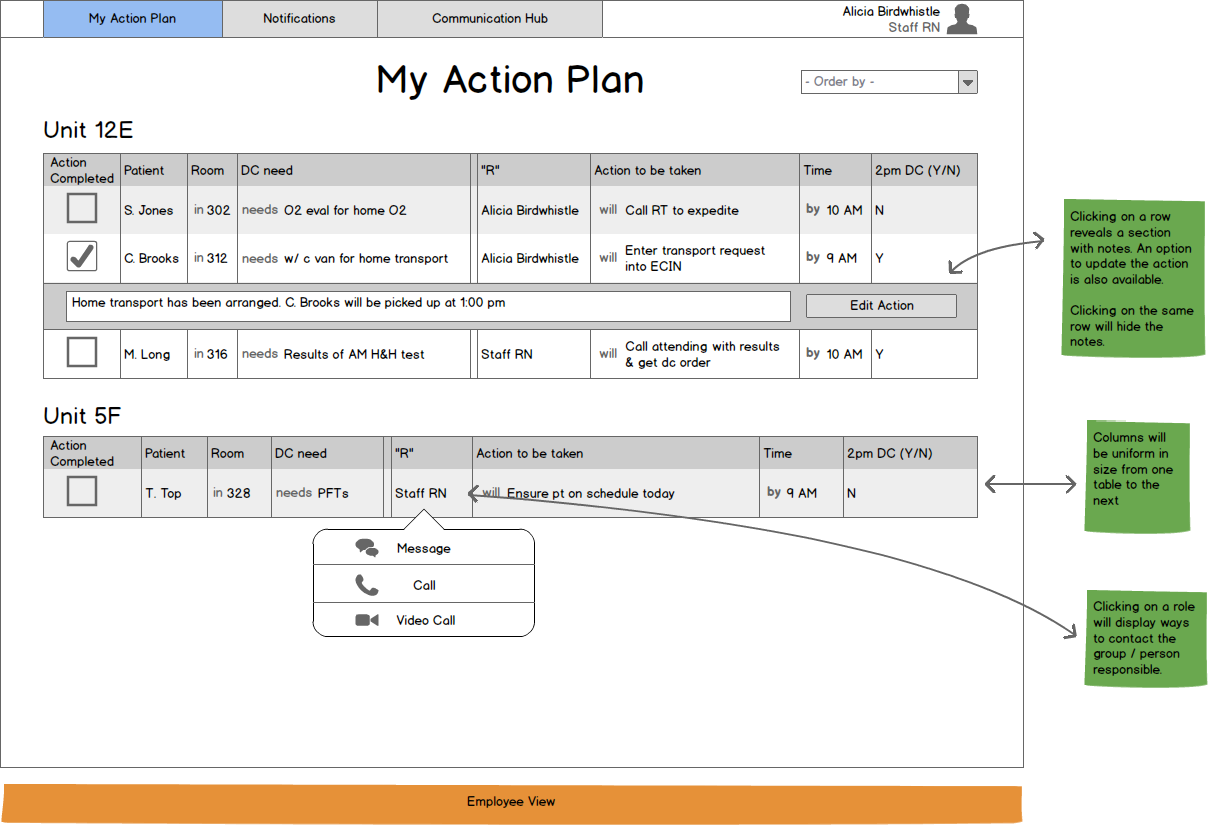


Figure - Employee's personal action plan



Figure - Communication hub

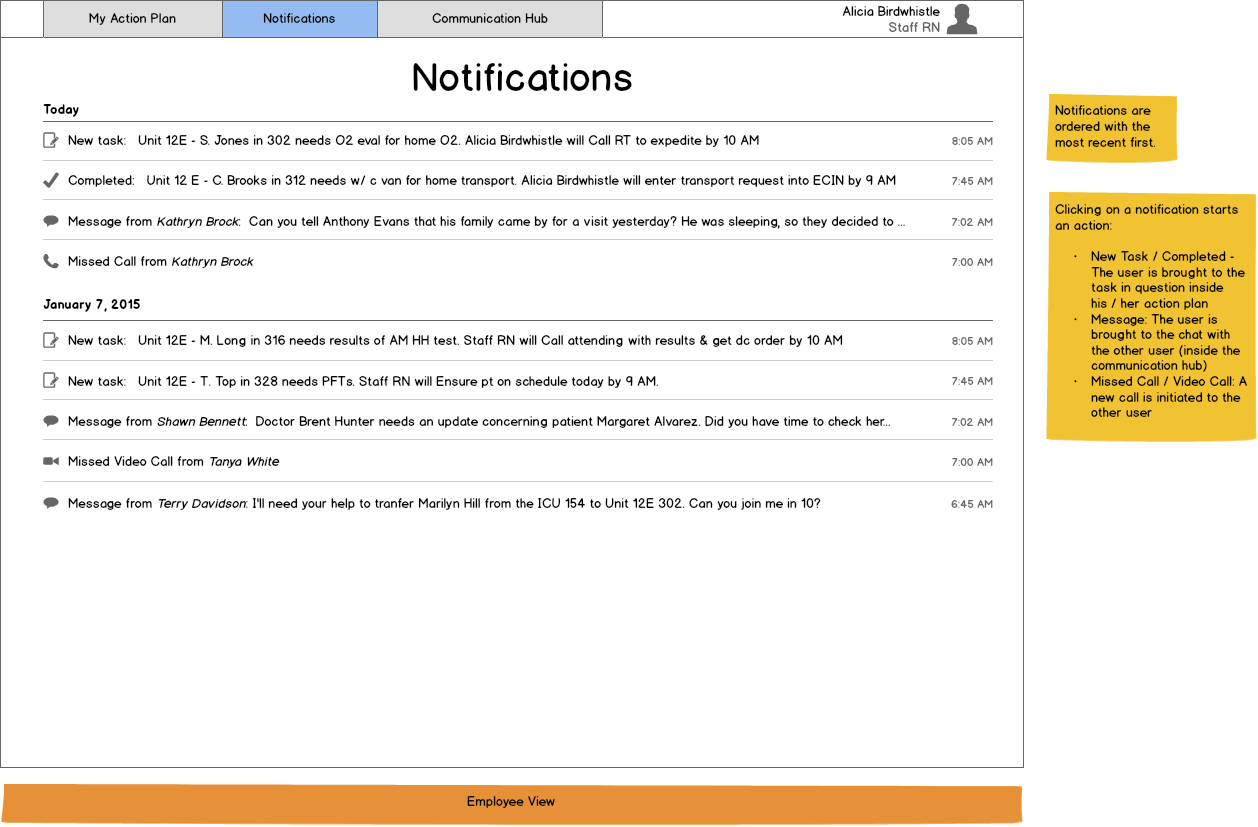


Figure - Notifications

# Architecture Description

The RTDC tool is client-server system. As illustrated in the deployment diagram below, three different clients will be supported: one for Android, one for iOS and one for the web. These clients will connect to two server components. The first is an Asterisk server which is at the core of the unified communications feature. The second is a Java server which hosts the remaining functions and interacts with the NoSQL data store known as Couchbase.



# Risk Assessment

## Engineering Challenges

Multiple engineering challenges will need be solved throughout the development of the RTDC project:

* **Cross-platform Support**: Developing a web interface and native apps for Android and iOS implies the use a variety of languages. It also extends the number of software versions to test against. Finally, the interface of the application will need to be adapted for touch input on a variety of screen sizes.
* **Network Operations:** There is no guarantee that the hospital network will be accessible from all rooms in the building. Therefore, connection lost needs to be taken into account. On the other hand, conflicting changes will need to be resolved when synching data with the database.
* **Ease of Use & Sturdiness:** The interface needs to be easy to use for users with no technical background. Furthermore, the application will need to be responsive, especially for network operations. Lastly, exceptions will have to be handled gracefully as the application should always be accessible.
* **Communication Framework Integration:** Taking advantage of Asterisk will require libraries which can communicate and exchange data with the framework.

## Organizational Challenges

Undertaking a project of this size presents two major organizational challenges, as described below:

* **Schedule Management:** The customer and team members all have different availability. Thus, meetings will have to be well planned to take efficient use of everyone’s time.
* **Centralization of Information:** Information pertinent to the project is spread out to a variety of sources such as websites, emails, text messages, etc. Presentations, documents, code and all other information should be accessible in one place.

## Deployment Challenges

Demonstrating the progress of the application to the customer presents a few challenges:

* **Hosting:** The web interface of the application will need to be hosted on a server accessible from the outside. Asterisk will also require hosting for the communications feature to work.
* **Mobile App Installation:** While Android offers a variety of options for distributing apps for free, iOS requires a paid license. For private distribution (outside the App Store), an Enterprise developer license is necessary. However, project files can be packaged and shared for demonstration using a Mac and the iOS Simulator (free of charge).
* **Integration with Existing System:** The RTDC project is a component of bigger platform known as the Next-generation Patient Flow Management System. Integration with the existing platform may prove difficult. However, it opens the possibility of adding new features such as automatic data input.

# Impact Assessment

## Legal Issues

The use of the application falls under Ontario’s Personal Health Information Protection Act of 2004 (PHIPA). In short, the collection of personal information requires consent and shall be limited to the strict minimum necessary. Furthermore, security safeguards need to be put in place to protect the information.

While the RTDC tool facilitates the execution of tasks towards a patient’s discharge, a discharge order is still required as per regulation 965, section 16 of Ontario’s Public Hospitals Act of 1990 (PHA).

The RTDC project source code will be the intellectual property of the student team listed in the *Project Title & Team* section of this document. The students retain the right to reuse any code as they please, and continue the development of the application once collaboration with the customer is over. Nevertheless, the customer will have access to the code to continue developing the system for production use. The customer may share the code with partners Openface / Health Now! for that purpose.

## Liability Issues

Offences to the Personal Health Information Protection Act can result in fines of up to $50,000 for a convicted individual, or $250,000 for corporations. Similarly, offences to the Public Hospitals Act can result in fines between $50 and $1,000 for a convicted individual.

## Societal Issues

The RTDC project has the potential to be very beneficial to society as it may help hospitals increase bed availability. Consequently, hospitals could see reduced wait times and patients will have a lower risk of health complications while waiting to be discharged.

## User Community

The intended user community, the hospital staff, may see an increase in productivity and efficiency in their work.

## Financial Impact

The design and development of the RTDC project will be done free of charge before the end of 2015. Afterwards, support, maintenance and updates through the RTDC team may be done at a fee (to be negotiated).

The customer or its partners Openface / Health Now! are responsible for providing the hardware on which the application will be deployed. This includes servers capable of running Asterisk and the RTDC application, as well as tablets. They are also responsible for the cost of an iOS Developer membership.

As previously mentioned, deployment to iOS devices requires a paid membership to the iOS Developer Program. A personal subscription for distribution through Apple’s App Store can be obtained for $99/year. However, private distribution requires an iOS Developer Enterprise subscription, available for $299/year.

# Project Plan

The project roles have been distributed as follow:

* **Project Manager:** Nicolas Ménard
* **Business Analyst:** Philippe Legault
* **Test Manager:** Olivier Clermont
* **Build Manager:** Jonathan Ermel
* **Architect:** Mathieu Fortin-Boulay
* **Lead Developer for Client Component:** Olivier Clermont
  + **Client Component Developer:** Nicolas Ménard
* **Lead Developer for Server Component:** Mathieu Fortin-Boulay
  + **Server Component Developers:** Jonathan Ermel, Philippe Legault

The project will take place over a period of 8 months. In the first phase (winter semester), the goal will be to develop the application itself. In second phase (fall semester), the application will be integrated with the Next-Generation Patient Flow Management System and the collection of data will be automated. The project manager will oversee the progress of the project and write the various project status reports. Here is a detailed overview of the progress planned for each iteration:

## Iteration 1 - Project Definition

The project definition is established, and the foundation for the project has been laid. In other words, Asterisk has been installed on a server, the basic code files have been created and the cross-compiling system has been configured. Moreover, the development of the Android client has begun. The project manager is responsible for the project definition document.

## Iteration 2 - Requirements Analysis

The Android interface is complete and mostly functional. Server components unrelated to unified communications have been implemented. The integration of Asterisk has begun and is well-off. Development of the web interface is also well-off. The business analyst is responsible for the analysis report.

## Iteration 3 - Demo & Quality Assurance Plan

Both iOS and web clients have been implemented. The notification feature has been added to the application and Asterisk is fully integrated into the system. A demonstration is made to the client. The build manager is responsible for the quality assurance presentation.

## Iteration 4 - Alpha & Architecture Design Report

Server component fully supports the Clinical Operations Object Model. The application is integrated into the Next-Generation Patient Flow Management System. The architect is responsible for the design report.

## Iteration 5 - Beta & Quality Assurance Report

The project is almost complete and automatic data input is implemented. This includes the capacity data from the Capacity Management Decision Support System and the demand data from the Demand Management Decision Support System. The test manager is responsible for the quality assurance presentation.

## Iteration 6 - Deployed & Final Report

The project is complete. The application has been packaged and deployed to the customer. The project manager is responsible for the final report.