

Fall  
2020

# SW Engineering CSC648/848 Team 2



MILESTONE 1 DOCUMENTATION

SFSU | CSC648/848

## Contents

1. Executive Summary .....	2
2. Personas and User Stories .....	3
3. Data Definitions .....	4
4. Functional Requirements .....	5
5. Non-Functional Requirements .....	7
6. Competitive Analysis .....	9
7. High-Level System Requirements.....	10
8. Team .....	11
9. Checklist.....	11

## 1. Executive Summary

During these unconventional and trying times, there seems to be a lack of availability in the physical therapy market to help facilitate the ease of acquisition of medical care as well as the administration of said care for those needing rehabilitation from injuries or surgical procedures. This is the gap that PorTal seeks to fulfill with its physical therapist dashboard. As the extant client-facing dashboard pairs users with registered therapists, our dashboard will provide the medical staff remote access to records, progress tracking, and vital interfacing options to aid in successful physical therapy and rehabilitation. Looking beyond the circumstances of the pandemic, the success rate of therapy is directly correlated to the ability for clientele to stick with the assigned exercise regimen. When this can be conducted properly in an asynchronous manner, it allows for greater flexibility for both the client and therapist while not sacrificing the overall quality of care provided.

Specifically, the portal will allow therapists to organize and view their patients, both current and tentative. Within each patient page, generating tailored exercise programs will be as simple as dragging and dropping exercises from a curated library of common exercises used in therapy. Beyond this, the ability to upload custom instructional videos will be simple and intuitive. For often used programs, or treatment of specific commonplace injuries, therapists will be able to create workout regimens and save them for easier deployment later. From the client-side, uploaded videos of exercises performed will be available to the therapists under each patient's profile for review and assessment of progress, to include feedback on individual videos. Messaging between patients and therapists will be available around the clock, with an adjustable email notification system to alert therapists to patient questions and concerns in a timely manner. Tracking metrics will be implemented so that therapists and administrative personnel can generate regular reports which detail progress of specific patients across multiple time scales as well as logging therapist's time investment on a per-client basis. Furthermore, it is vital that the web application secure access to medical records in a manner that protects all users and complies with HIPAA regulations.

Aside from bridging the gap for remote physical therapy, something that is currently missing in the medical services space, our web application will seek to provide additional tools to assist physical therapists in assessing clients that increases efficacy and accuracy which may be a concern for therapists accustomed to in-person evaluations. The ability to draw on screenshots of client-submitted videos, while providing joint-angle calculations, allows for quantitative measure of range of motion and progress tracking. Our team is also investigating the possibility of implementing open-source AI (artificial intelligence) tools to provide body position in videos, either via real-time or separate processing, that can speed up this necessary portion of physical

therapy. This is largely contingent on the required resources to perform said calculations while minimizing the outlay costs to the physical therapists using the platform. That being said, our primary goal is creating a web application that functions across a variety of devices and web browsers seamlessly so as to maximize our total addressable market.

Our team is a diverse group of SFSU computer science students working to prototype a vertical slice of a full-fledged web portal to be utilized by physical therapists. This project is specifically tailored to Netic, a Berkeley start-up. After assessing their needs and goals, the following is an outline of our understanding and process to achieve such a product.

## 2. Personas and User Stories

### Personas:

- 1. Therapist with troubles tracking patients** – Jamie is a physical therapist that has many issues with keeping track of patients in between the standard weekly or biweekly appointment. As many of her patients have extra exercises to perform on their own and sometimes have questions that are delayed until the next appointment, this often hinders the rehabilitation of her patients.
- 2. Therapist with insufficient access to community** – Greg has a physical therapist office in a rural area that often requires his patients to drive long distances that often conflict with their schedules and/or lifestyles. This results in many missed and rescheduled appointments. This also has prevented his practice from growing because some of the patients that need help, simply cannot find the time to attend sessions in person.
- 3. Therapist requiring faster assessments** – Alice has a lot of patients in her practice and spends a lot of her day, outside of appointments, reviewing progress and assessing patients on a continual basis. This has prevented her from taking on more clientele because of the associated workload on the backend. Provided more advanced tools, Alice and her colleagues could expand their practices and help more people concurrently, helping to clear the significant backlog of new patients waiting to be seen.
- 4. Administrator requiring performance metrics**– Bob is a hospital administrator that oversees the physical therapy department. He needs to keep track of individual therapists and their engagement with patients on a granular level but their system currently has no way of documenting this.

### Use (Persona application in parenthesis):

Our application seeks to provide a portal for physical therapists that closes the gap between in-office appointments and patient self-care at home (1,2). The ability to upload videos of exercises conducted when the patient has time (2), also provides the ability to expedite patient tracking and leverage computational assets for reviewing patient progress (1,3). This portal also allows for complete assessment of patients via asynchronous correspondence, which helps to expand the reach and availability of needed care to underserved areas and help those that may not have the resources or time to regularly attend in-person appointments (2). Often times, there are questions patients would like to ask but it often occurs outside of normal business hours or the therapist is currently busy and cannot come to the phone. The messaging system in the app will rectify this by allowing interactions at any time of the day, suiting the patient's and therapist's schedule in a more fluid way (1,2). The application will also provide tracking of time the therapist interacts with each patient profile, whether they are responding to messages, developing exercise plans, or assessing uploaded videos from the patient. This allows for therapists and their supervisors to extract meaningful metrics to evaluate themselves and staffing needs (4).

## 3. Data Definitions

1. Unregistered User: A user who does not have an account or not logged into the system. Their usage of the website is limited to the landing page and pages explaining the app.
2. Registered User (Therapist or Patient): A user who has an account and is currently logged in. They use the website to access their personal information pertaining to their physical therapy.
3. Administrator: A user who has access to performance metrics for physical therapists. They will use the website to see how much time each therapist is spending with each of their clients, as well as, running reports over different time scales.
4. Patient: A user that has an account and logged in via the patient-side portal. They do not interact directly with our physical therapy portal, but the communications and video uploads from patients will be directly accessible by our dashboard for physical therapists to see.
5. Therapist Dashboard: A landing page upon login that displays any relevant alerts, messages, and a list of current and prospective patients. Links to all important pages will be easily identified and labeled.

6. **Directory:** Page listing of all therapists currently enrolled in the system and their physical office location, if applicable.
7. **Exercise Library:** A collection of exercise descriptions and **videos** it can be used by the therapist to create or edit custom workout regimens to save and later assign to one or many patients.
8. **Patient Video Library:** A collection of patient-submitted videos that is available for physical therapists to reference as needed when marked as sharable by patient.
9. **Patient Profile:** Page detailing the patient's injuries, current exercise regimen, and uploaded videos for assessment.

#### 4. Functional Requirements

Functional Requirement No.	Function Requirement Description (Owner, Priority)	Applicable Personas
FR 1	PTs and User able to create their profile with relevant details. (Admin, User, 1)	1-3
FR 2	Admin User able to delete or modified PTs User profile. (Admin, 1)	1-3
FR 3	PT able to view all the current patient's profiles. (Admin, User, 1)	1-3
FR 4	PT able to view all the prospective patient's profiles. (Admin, User, 1)	1-3
FR 5	PT able to upload training video into exercise library for use in creating regimen. (Admin, User, 1)	1-3
FR 6	Patient able to select privacy setting for uploaded video for sharing or private use. (User, 1)	1,3
FR 7	PT able to delete training video in the patient library, upon request of patient. (Admin, User, 1)	1-4
FR 8	PT able to pick training video from the library for customized exercise. (Admin, User, 1)	1-3
FR 9	PT able to drag and drop exercises from the library. (Admin, User, 1)	1-3
FR 10	PT able to add customized exercise plan for specific patient. (Admin, User, 1)	1-3
FR 11	PT able to assign existed exercise plan for different patient. (Admin, User, 1)	1-3
FR 12	PT able to view current patient home exercise plan. (Admin, User, 1)	1-3

FR 13	PT able to view the patient video history (Admin, User, 1)	1-3
FR 14	PT able to draw line directly on the video. (Admin, User, 3)	1-3
FR 15	PT should be able to calculate angles automatically from the patient's video. (Admin, User, 3)	1-3
FR 16	PT should be able to estimate the pose from the patient's video. (Admin, User, 3)	1-3
FR 17	PT able to give feedback to patients by text format. (Admin, User, 1)	1-3
FR 18	PT able to give feedback to patients by video format. (Admin, User, 1)	1-3
FR 19	PT able to share videos of exercises with other's PT. (Admin, User, 2)	1-4
FR 20	PT able to share videos of exercise with other's patients, when permission is granted. (Admin, User, 2)	1-4
FR 21	Patient able to ask question via text communication. (Admin, User, 1)	1-3
FR 22	PT able to response question via text communication. (Admin, User, 1)	1-3
FR 23	PT able to receive email notification. (Admin, User, 1)	1-3
FR 24	PT able to change the frequency of the emails. (Admin, User, 1)	1-3
FR 25	PT able to track duration of each patients. (Admin, User,1)	1-4
FR 26	PT user able to see indicator on each patient's user profile. (Admin, User, 1)	1-3
FR 27	PT able to generate report on PT's activity with patients by day, week or month. (Admin, User, 1)	1-4
FR 28	Patient able to create patient's user profile. (Admin, User, 1)	1-3
FR 29	Admin able to delete patient's user profile. (Admin, 1)	1-3
FR 30	Patients User able to upload video to corresponds PTs only. (Admin, User, 1)	1-3
FR 31	PT and Patient messaging should be properly encrypted for confidentiality (User, 1)	1-3

## 5. Non-Functional Requirements

Operation Group - Describe the user's needs for using the dashboard's automated functionality

Access Security: System safeguarded against internal and/or external intrusion.

- Password protected areas authenticate and authorize clearance in accessing patient data

Accessibility: Software extends user base across range of services and goals

- Remote and asynchronous therapy assists in providing medical care beyond traditional bounds and areas.
- Currently only planning for support in English language

Availability: User access during "normal operating times"

- Dashboard's asynchronous platform operates on all modern browsers
- AWS server up time averages greater than 99.99% if required by customer

Confidentiality: Degree software protects sensitive data and allows authorized access to data

- Medical system's patient records must follow all state and federal laws
- US Health Insurance Portability and Accountability Act

Efficiency: Extent software handles capacity and response time

- Processor capacity and storage space available on system during peak load
- System restart cycle
- Notifications and report summaries

Integrity: Data maintenance and accuracy without corruption

- Changes made to database are recorded routinely and backed up

Reliability: System performs functions without fail

- Point of interaction delays and/or failure recourse

Survivability: System function and recovery

- Simulation of > 100 concurrent users not tested due to constraints of student funded AWS framework
- Dynamic scaling, vertically and horizontally, available for production environments
- Data recovery in roll-back
- Data redundancy ensured via scheduled backups

Usability: Ease of use intuitive and effective

- No training is required for product use
- User interface (UI) self-explanatory

Revision Group – User needs addressed and perceived as modifiable

Flexibility: Software is adaptable to different environments and configurations of user expectations

- Universal usage (preferred language)



- Ability to add new user information without affecting earlier forms

Maintainability: Ease of fault discovery in software

- System developed in popular MERN software stack, using MongoDB, Express, React, and NodeJS
- Software testing shows fixable problems in compartmentalized structure
- Rolling deployment minimizes required downtime to update application

Modifiability: Cost effective development, deployment, maintenance, modifications and updates

- New product features can be added to source code and tested in less than a week
- Function calls are NOT to be nested more than two levels deep

Scalability: Degree system is able to expand capabilities upward and outward

- Report, text and upload duration limitations

Verifiability: Extent needed to prove system will function as intended

- Test panel for product compliance
- Safety regulations
- System infrastructure and automation
- System updates follow quality assurance

Transition Group – System upkeep, including hardware, software and compatibility

Installation: N/A

- Browser based dashboard

Interoperability: Extent of software interfacing with other systems

- HTML/JavaScript browser interface
- Universally recognizable pictures, icons and symbology
- Local device supports data interface, protocols and security solutions

Portability: Software transferable between environments

- UTC – Universal Time Coordinated
- Targeted market
- Developed for different OS platforms
- Supports minimum color, speed, storage and monitor standards/benchmarks

Reusability: Software ease of conversion between systems

- HTML guidelines and standards
- Client device requirements met
- Templated codebase to provide easily extensible functionality

## 6. Competitive Analysis

Competitor features	Our features
Apps specific to body part/injury <ul style="list-style-type: none"><li>Someone with a spinal injury may not care about knee-related therapy</li></ul>	One app for all physical therapy <ul style="list-style-type: none"><li>All information lives in one place. A physical therapist doesn't need to upload information more than once.</li></ul>
Interactive 3D models <ul style="list-style-type: none"><li>Someone has to pose the 3D model. Human error could make the model's pose slightly off, ruining the effect of the therapy.</li></ul>	Videos of poses and exercises <ul style="list-style-type: none"><li>We are taking videos and using AI to trace a skeleton over the human body. We allow drawing/tracing onto videos to allow notations by a therapist.</li><li>Patients can upload videos for their therapist to review.</li></ul>
Methods for handling "assignments" <ul style="list-style-type: none"><li>The therapist can track what exercises the patient has done, and track their assignment progress.</li></ul>	Methods for handling "assignments" <ul style="list-style-type: none"><li>We are allowing therapists to post assignments, and show when their patient has completed it.</li><li>We may include notifications to the therapist when the patient has submitted.</li></ul>
Allow the therapist to send exercises to the patient directly <ul style="list-style-type: none"><li>Like sending a playlist, so the patient does not have to look up specific exercises and hope it's the same one.</li></ul>	Therapists can assign exercises to their patients. This is covered in the "assignment" handling.

Most apps seem to be "one or the other", either focused on giving the patient as much information as possible but not giving therapists a way to work with them directly, or giving therapists lots of control and ways of monitoring their patients but without providing the patient with ways of handling information or examining exercise routines. We are seeking to have the best of both worlds. We are focused on giving the therapist the most that we can, but without forgetting about the patient.

## 7. High-Level System Requirements

- I. Frameworks
  - a. Node.js
    - i. Version 4.11.0
  - b. React.js
    - i. Version 16.13.0
    - ii. Supported Browsers: all modern browsers i.e. Chrome, Firefox, Safari; Internet Explorer 9, 10 and 11 **requires** polyfills
  - c. Phonegap
    - i. Version 9.0.0
  - d. Material-UI
    - i. Version 4.11.0
    - ii. Supported Browsers: Chrome (version 49 or greater), Firefox (version 52 or greater), Safari (version 10 or greater), Internet Explorer v.11, GoogleBot
- II. Client-Server Frameworks
  - a. Express
    - i. Version 4.17.1
- III. Backend
  - a. Framework
    - i. Apache Spark
      - 1. Version 2.4.0
  - b. Database Frameworks
    - i. MongoDB
      - 1. Version 4.4.0
  - c. Host
    - i. Amazon Web Services (AWS)
      - 1. Services: EC2, EBS, Elastic IP
  - d. Hosting OS
    - i. Ubuntu Server
      - 1. Version 20.04 (Focal)

## 8. Team

Team Lead: Jarett Koelmel

Front End Lead: Chiu Wong

Front End Dev: Michael Canson

Front End Dev: Paul Borst

Backend Lead: Peter Hu

Backend Engineer: Brooke Porter

GitHub Master: Eric Chen

Scrum Master: Jarett Koelmel

## 9. Checklist

Team meeting scheduled outside of class: DONE

GitHub Master chosen: DONE

Team agreed upon using listed SW tools and deployment server: DONE

Team ready and able to use the chosen back and front-end frameworks and those who need to learn are working on it: DONE

Team lead ensures that all team members read the final M1 and agree/understand before submission: DONE