



Tecnológico
de Monterrey

LABORATORIO DE APLICACIONES WEB

KASSNA PROJECT

*ITZEL MOSSO GUERRERO A014223891
ARTURO SALINAS ORTÍZ A01835189
LUIS SALOMON FLORES UGALDE A00817485
MATTHIAS BRAEUER A01679521
MIGUEL ANGUEL BANDA A00816758*

31 DE AGOSTO DEL 2018

I. Introduction

This document contains the specification for the project that will be developed by the Red Team during the semester August-December 2018, the project is for Kassna, an startup that works as a massage service at home. The main content of the document is the agreement between Sergio Larraguibel that from now on it will be called **The Client** and the development team (Red Team). The document also includes the schedule of audits and the scope of the progress projected for this semester.

The project to be developed is a web application that will be updated, the application is running in the main website of Kassna, this site is managed most of the time by the client, in this site he receives all the massage requests then he looks for an available massage therapist and assign it to the current request and so on. When a request is assigned or declined the client receives an email notification and the appointment for the massage is done. Our update will make able to have agents that works for Kassna, an agent is a person that can make an appointment for many clients, they usually are the front desk of hotels, gym or academies. The agents will have a rewards system, where for each appointment they complete they receive points that they can change for money or massages.

Considering these precedents, the project will represent an update of the actual website, keeping and getting better the functionality, following the design and architecture patterns and using the technologies that the web application is using.

II. Project Objectives

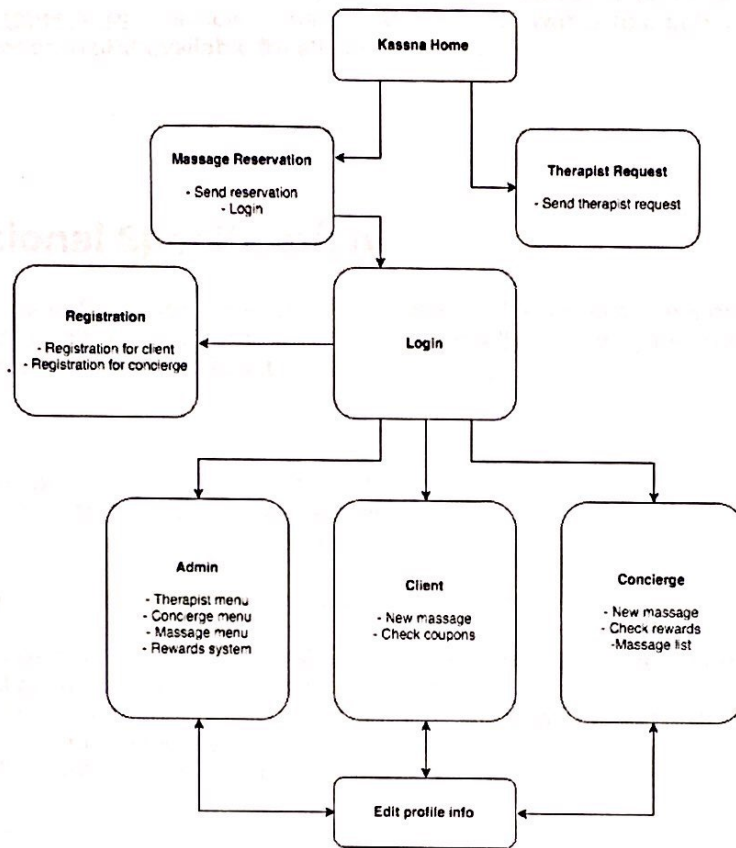
The purpose of the project is to develop an application for external personnel to Kassna or concierges to make requests for massages on behalf of a client. In this platform the concierges can capture all the information regarding the massage so that Kassna staff can receive it and process the order more quickly, in addition to making the charge more efficient. Also a record of the registered orders of each concierge will be kept in order to implement an incentive system that promotes the sale.

III. Success Criteria

This system seeks to increase sales by 15% by improving communication and incentive channels in hotels, gyms or through concierges. With this new system, we will be capable of reach many clients as well as getting more recommendations from new crews. Also, a new reward system has to be implemented as for getting a different environment.

Both the increase of sales and the brand new reward system will be the most beneficial, thus, crucial part for accepting this project.

IV. Site Map



IV.I Content of the web application

Based on the sitemap structure, these are the functionalities for each web page(view):

- **Kassna Home:** This is the web site of the client, you can do a message reservation, login or send a therapist request there.
- **Message Reservation:** Here any person can do a reservation and if you have an account, you can login and access to the system.
- **Registration:** Here a possible client or concierge can send a registration request that will be approved or not by the admin.
- **Login:** In this web page any user can give their credentials to access to the system.
- **Admin:** This screen is loaded when an admin logs in.
 - Therapist menu: Here the admin can see everything about therapists, requests, assign, etc.
 - Concierge menu: Here the admin can see everything about concierges, request, message reservation, etc.
 - Massage menu: Here the admin can see everything about massages, new request, old and new massages, etc.
 - Rewards system: Here the admin can check and modify the rewards system for every concierge.
- **Client:** This view is the one that loads when a client logs in.

- New message: Here the client can request a new message.
 - Check coupons: Here the client can see if he has positive balance
- **Concierge:** This view is loaded when a concierge logs in.
 - New message: Here the concierge can request a new message.
 - Check rewards: Here the client can see his points and rewards.
 - Message list: The concierge can see the message request he have done.
- **Edit profile info:** This view is loaded when a user wants to edit his info, like phone number or mail, it available for all kind of users.

V. Functional Specification

Here are the main functions for each of the roles in the system. For simplicity, here are listed all the possible actions a role can do, but internally, they may be complex, including dependencies and other specific actions.

Any person:

- Will be able to send an agent request.
- Will be able to send a therapist request.
- Will be able to reserve a massage.

All type of users: This use cases applies to all kind of users

- He will be able to login using valid credentials (username and password)
- He will be able to logout to finish the session.
- He will be able to modify his credentials(username and password) once logged in.
- He will be able to recover his password.
- He will be able to edit/update his profile.

Types of users:

Agent:

- He will be able to send a massage request.
- He will be able to see his request history.
- He will be able to cancel a request.
- He will be able to edit a request.
- He will be able to see his rewards.
- He will be able to see his profile.
- He will be able to see the status of a massage request.

Admin:

- He will be able to accept an agent register request.
- He will be able to decline an agent register request.
- He will be able to see an agent register request.
- He will be able to edit rewards.***
- He will be able to see agents list.***
- He will be able to delete agents from the list.
- He will be notified when a new register request(agent, therapist) is sent.

Client:

- A lot of cases of use are not here because the system already has it, these are the ones we are going to develop.

The project is built with MeteorJS, as a full stack framework. Using MongoDB, we will create the new environment for running the project, but attached to the existing base.

A file structure for the project's directory can be found in the GitHub repository for Kassna, it uses the file structure that comes with meteor-base.

The following is the general structure of the project as an ERD diagram, showing the main actors and relationships to build the DB.



VII. Content Plan

The Client

- Responsible of providing information about the background of the enterprise
- Responsible for verifying, validating and accepting all the new improvements in every sprint or meeting.
- Responsible for informing new possible changes to the project scope, as well as clarifying ambiguities from components and feature usages.
- Responsible for giving feedback for every deliverable.

The Team

- Responsible for delivering a complete functional project as planned in this document
- Responsible for showing all the new improvements through the established timeline
- Responsible for maintaining a good version control during the development of the project

The Professor

- Guidance for the developer team as for to cover the course scope, technical questions or course requirements, as well as for delivering a good project for the client
- Resolve questions and issues between the team members and/or the client
- Reviewer of the overall progress and give feedback.

Relations

The team works directly with the client, delivering new features for the client and making corrections of previous work if all the involved persons agree

VII.I Version Control

GitHub will be the platform in which the version control of the project will take place. The project from Kassna already has a repository in [Github](#), this would be the production branch, as well as development. The working environment for this project should be done in a branch specifically for it or a completely new repository.

VII. II Responsibilities

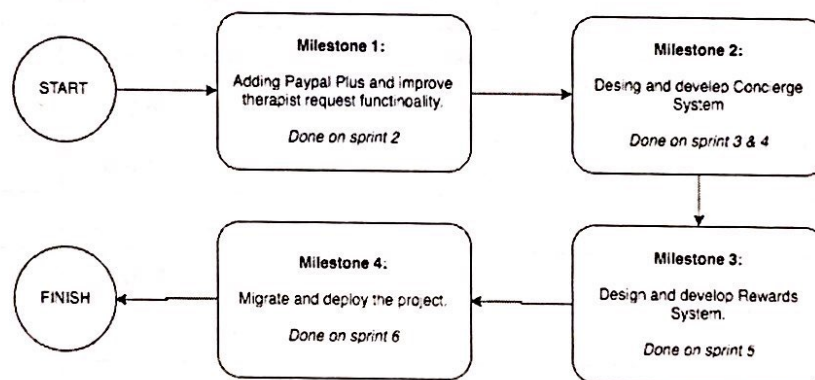
Name	Role
Sergio Larraguibel	Client
Miguel Banda	Team Leader / Web Developer (Front End) / Design
Arturo Salinas	Web Developer (Front End) / Documentation

Luis Salomon	Web Developer (Back End) / Design
Itzel Mosso	Design / Web Developer
Matthias Bräuer	Web Developer (Back End) / Quality Manager

VII.III Timelines and Milestones

This is simply a list of the meetings that are going to be held between the developing team and the client to be in tune of the current development of the project.

Timelines and milestones:



Priority	Sprint 1: September 4 - 18
	Know the project
	Learn about the frameworks
	Setup testing environment
	Create a testing database
	Design the views for the next new features
	Setting the pushing rules, code style, naming convention, etc.
	Sprint 2: September 18 - 25
	Paypal plus implement.

	Refactor therapist application form.
	Email notification for Admin when new therapist request.
	Check DB compatibility and refactor if needed(therapist table)
	Sprint 3: September 25 - October 9
	Design Concierge system
	Develop Concierge
	Email for Admin from Concierge application and to Concierge for confirmation(Account)
	Application form for Concierge
	Concierge Request for massage form
	Sprint 4: October 9 - 23
	Admin panel with new tab for Concierge
	Admin panel with new tab for Concierge requests
	Email notification for Admin about new massage request
	Edit Concierge from Admin panel
	Accept Concierge Request from admin panel
	Usability test
	Sprint 5: October 31 - November 13
	Design Reward System for Concierge
	Develop Reward System for Concierge
	Reward system available for admin and concierge
	Usability test
	Change language configuration of the whole site
	Sprint 6: November 14 -November 20
	Deployment of the application
	Migrate DB if needed
	Elaborate a user's manual
	Training to Kassna's personel

VIII. Testing Plan

Important: Test-Cases are defined during the development process when user stories and the corresponding acceptance criteria are defined.

Each user story contains acceptance criteria which results into a acceptance test.

User stories and acceptance criteria/tests are defined in the corresponding Github issue.

- Iterative Tests:
 - Unit-Tests
 - To test each unit independently for its correct behaviour
 - Integration-Tests
 - To test each unit within the application
 - Static Code analysis
 - To ensure a clean and consistent code
 - Acceptance-Tests
 - To test each feature manually if it fulfills all acceptance criteria (this is done by one of the developers that did not develop the feature that is tested and shortly by everyone in the sprint reviews)
 - The client has to do an acceptance test every once in a while to confirm correct behaviour of the application or to report bugs / unwanted behaviour (Bugs and unwanted behaviour have to be reported in the [Github](#) repository as a issue with a clear description of the issue and when it occurs).
- Usability tests
 - To ensure a good user experience usability tests will be done at the end of the project before the deployment to the production server.
 - Each test consists of a group of 3-5 people where each of them will try to fulfill certain tasks and speaks out loud all the upcoming issues that prevents that person from succeeding.
 - After each group of tests the application to fix all issues that came up.

IX. Appendix

Meeting Notes

First meeting

- The team meets the client
- Explanation of the most important requirements to be develop
- Explanation of the actual way of work of the client
- First aproach of a general idea of the project.

- Filled a small document as a proposal for the professor to work during the semester.

Second meeting

- Established objectives for the project specification document.
- Established the success criteria
- Proposal of the deliverable project as a whole
- Explanation to the client of the most important components to be created
- Planned the meeting dates and way of work (based on sprints)
- Planned the technologies to be used during the development

Design Concepts

Prototyping

Assumptions

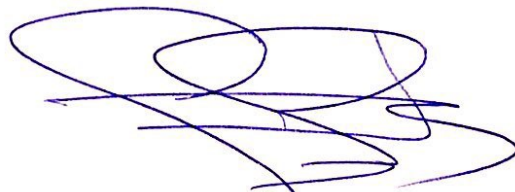
The need of a testing database. Until the team gets into actual development and the actual design of the concierge, and what test cases it needs to do to fulfill the requirements, this testing environment is more than optional. (It is necessary to run a local or external database to see if the creation and edit of the tables is done correctly though)

Glossary of Terms

Concierge and Agent : refer to the new module that is asked for.

X. Client and Team Leader Signature Page


Miguel Angel Banda Rdz.
Team Leader


Sergio Lammarel Beers
Client