**Software Project Management Plan**

Group 3

**Part 0: Front Matter**

**Revision Sheet**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Updated by** | **Update comments** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Scope and purpose:** To allow individuals to reserve space at unused school venues with ease. The process of reserving the venue, inviting people to the event, and buying tickets to the event will all be simplified.

**Part 1: Introduction**

**Project Overview:** This project is to create an online resource that will allow individuals to reserve unused school venues (auditoriums, gyms, etc) at the discretion of the school management. The online resource will be user friendly and will allow to be used on mobile devices. The user will be able to look at the school and venue they want to pick. Once they pick the location, they will be able to set the total amount of tickets available and the price of each ticket. Depending on the location, a fixed price will be made for reserving the spot (location and duration of the event will reflect the price). Once the management of the school agrees to the event, the user will pay, and the information of the event will be placed on the website’s database. Other people will be able to know of the event through social media.

**Project Deliverables:**

|  |  |  |
| --- | --- | --- |
| **Deliverables** | **Delivery Date** | **Location** |
| Project Requirements | February 16, 2017 |  |
| Use Cases & Sequence Diagrams | February 23, 2017 |  |
| Project HLA & Class Diagram | When will we submit |  |
| Repo Setup | March 14, 2017 |  |
| Website Host | February 20, 2017 |  |
| Web development Progress / Previews | Up until April 30, 2017 |  |
| Functionality Testing/Debugging | Up until May 11,2017 |  |

**Evolution of the SPMP:**

* Delivery Dates - May be updated based upon progress
* References - Updated when applicable
* Definitions & Acronyms - Updated when applicable
* Organizational Structure/Project Responsibilities - Can change when members are assigned different roles or leadership positions
* Organizational Interfaces - Software being used can change given difficulties with usage and implementation
* Managerial Process - Priorities can change based upon factors such the customer or constraints
* Technical Process - Will be updated as project is developed

**Reference Materials:**

|  |  |
| --- | --- |
| [www.os-templates.com](http://www.os-templates.com) | Retrieved CSS template for website |
| [www.biz.nf](http://www.biz.nf) | Acquired free website host |
| [www.github.com](http://www.github.com) | Environment for version control |
| <https://www.mysql.com/products/workbench/> | Database Creation/Management Software |

**Definitions and Acronyms:**

HLA - High Level Architecture

Repo - Repository

POC - Point of Contact

HTML - Hypertext Markup Language

SSL - Secure Sockets Layer

CM - Configuration Management

CSS - Cascading Style Sheet

PSD - Photoshop Document / Design

SQL - Standard Query Language

PHP - Hypertext Preprocessor

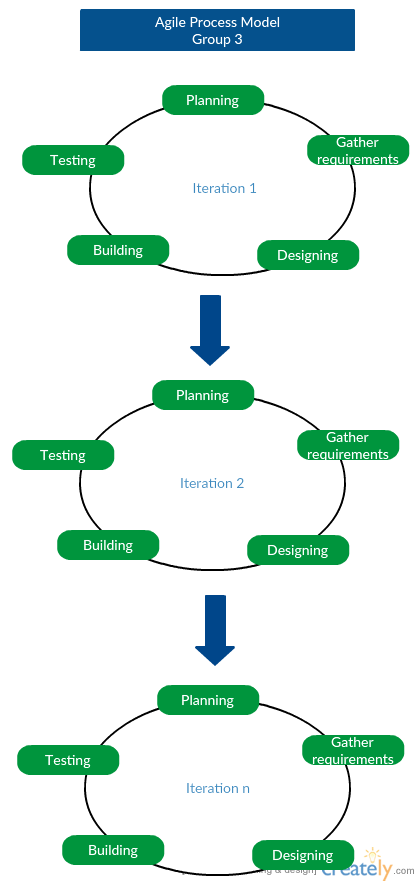
HTTPS - Hypertext Transfer Protocol

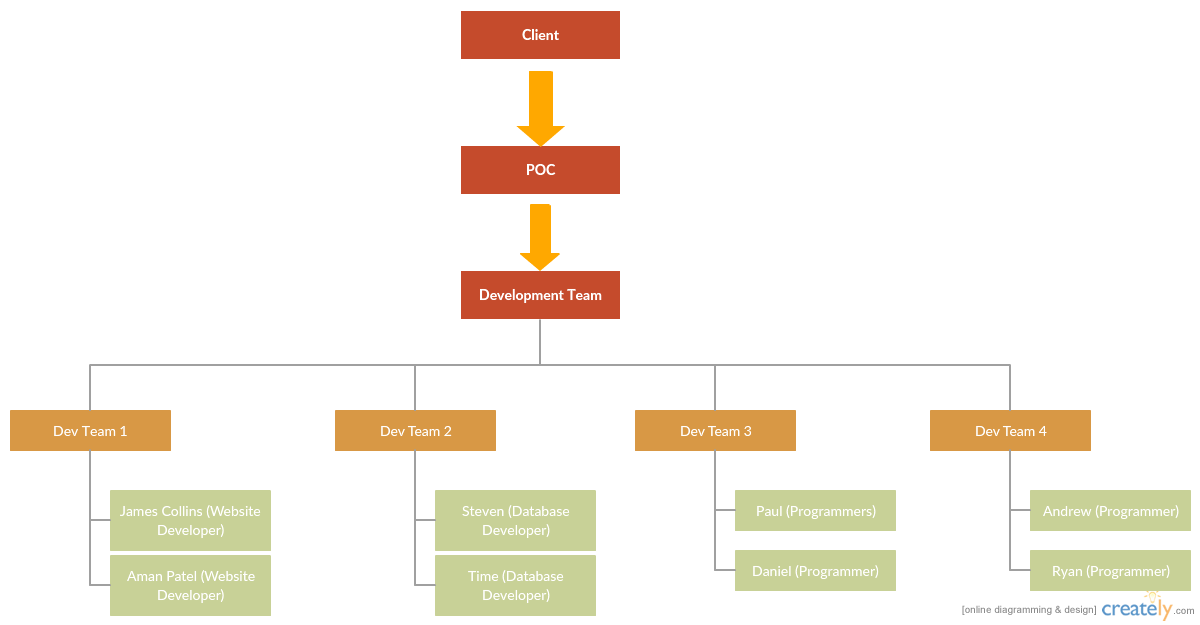
CSA - Configuration Status Accounting

CI - Configuration Identification

**Part 2: Project Organization**

**Process Model:**



**Organizational Structure:**

**Project Responsibilities:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Title** | **Group 1** | **Group 2** | **Group 3** | **Group 4** |
| **Website design** | **✓** |  |  |  |
| **Database Management** |  | **✓** |  |  |
| **Payment Engine** |  |  | **✓** |  |
| **Website information** |  |  |  | **✓** |
| **Social Media** |  |  |  | **✓** |
| **Mobile Friendly** |  |  | **✓** |  |
| **Domain management** | **✓** |  |  |  |

**Part 3: Managerial Process**

**Management Objectives and Priorities:**

During the development of this project the management philosophy will focus of a test-driven design. We will set small goals and test the project repeatedly until these goals are satisfied and the project is ready to evolve further and meet a new set of goals. This allows the client to view our forward progress and the group to stay organized by assigning smaller tasks to each team member. The team will be setting deadlines for each version of the project to ensure one version of the project does not take too long to complete. For requirements, the team will focus on attributes of the software that will generate the most revenue for the client. Allowing users to search through as many properties as possible with relative ease is the most important because it will increase the chance of a user finding properties that fit their needs, and in turn increase income. It is expected that there will be a technical overview in March 2017, as well as a working prototype in May 2017.

**Assumptions, Dependencies, Constraints:**

Assumptions:

* Facilities used primarily on the weekends during the school year
* Facilities will also be used during the summer
* Facilities will only hold respectable events (birthday parties, circus, memorials, etc)
* Social Media will be used to notify individuals of bookings, along with emails
* Fees will be determined by the event, amount of people attending, time, refreshments, and usage of the schools other resources
* Thorough background checks on all individuals/parties accessing school facilities
* Events may be scheduled for more than one day at a time, and events can also be scheduled ahead of time

Constraints/Dependencies

* Events cannot conflict with school related activities, or prior booked events for facilities
* Facilities other than the ones rented can not be accessed (Ex. Football field is rented. School’s other facilities cannot be accessed).
* An email confirmation will be sent to ensure parties are accountable for payments that must be accepted at or prior to the time of booking and validation.
* High Schools reserve the right during the semester to cancel/rescind events based upon certain circumstances that may arise
* Acts of nature will depend on the damage, and circumstances of the event. Money in this case can/will be refunded unless both parties respectfully, and amicably decide otherwise

**Risk Management:**

Identifying Risks**:**

· The client making late changes to the requirements

· The client shortening the completion deadline

· Client has unrealistic expectations about the project

· Client has issues about early versions of the project

Staff Risks

· Lack of communication

· Inexperienced team members

Planning risks

· Time and other resources are not used effectively

· Progress is not well monitored

· Project milestones are not well defined and reached

Project Size

· Team realizes the project is larger than expected and cannot complete it in time

The team will keep track of client risks the POC for the team will maintain consistent contact with the client and ensure they have good insight on the evolution of the project prototypes. Staff risks and planning risks are tracked and lowered by choosing an experience project manager and ensuring all team members have access to the group communication medium.

**Staffing Plan**

The team for this project consists of eight members, each of which fulfill roles that range from software engineers and web developers, to client relation managers.. The project will also need maintenance as long as it is in use by one or two crew members.

**Monitoring and controlling mechanisms**

The POC for the team reports all progress to the client each week up until the project deadline. The team tracks the project progress through the use of the software Github.

**Part 4: Technical Process**

**Specify the methods, tools and techniques to be used on the project:**

**Methods/Tools/Techniques:**

The website will be implemented using a hosting service and template from [www.os-templates.com](http://www.os-templates.com), complete with CSS, HTML and PSD. This template will the be front-end of the application. Back-end services will be implemented via SQL/Java. The code itself will go through continuous edits & refactorings through a version control application, GitHub, shared among all team members.

Purchases will go through a front-end credit card validation before going to a third-party authentication service. The third-party service will already have the necessary security measures such as HTTPS and SSL certificates in place. The third-party service may change depending on ease of use for the client and ease of integration in the application.

The list of potential rentals will be databased by address, corresponding school, price, and availability in SQL.

**Describe the documentation plan:**

We will document as we code, recording who wrote/edited a line of code on what date/version number.

**Project Support Functions:**

**Plans to ensure quality assurance:** Alpha Testing, Beta Testing, Monitoring after launch, changing as needed

**Configuration Management Plan:**

* **Scope**: To limit the impact changes may have on the entire system
* **Organization Role**: A CM leader will be appointed to oversee all CM activities. He will receive all change requests, and will make any final decisions regarding those changes, including which software engineer will carry out approved changes.
* **Identification**: A CI document will contain the breakdown of subsystems and how they  
  are interrelated. Any approved changes will be returned to the software  
  engineer with an change approval sheet, a listing of all possible affected  
  subsystems, and any additional information the software engineer may  
  need before he begins changing code.
* **Configuration Control**: Software engineers will submit a change request to the CM leader. The CM leader will then analyze the request, using the CI document, the project design document, and the current prototype of the software. He will base his decision on how severely the change will impact the entire system and, more importantly, on the corresponding subsystem. Once his decision has been made, he must submit the change to the software engineer of his choice, as well as updating the CI document to accommodate the change, and the CM library to record the change request and decision.
* **Version Control:** Whenever the system or a subsystem is updated, the program build  
  number (version number) will be updated to reflect the change. The version number will follow a standard x.x.x input mask (for example, version 1.2.7), with each digit place corresponding to an increasing severity of change. The hundredths place (x.x.x) will reflect very minor changes to the software. The tenths place (x.x.x) will reflect more  
  substantial software changes. The ones place (x.x.x) will reflect severe changes to the software.
* **Configuration Status Accounting (CSA):** A draft of the changes will be sent to the client. Upon approval, the final change document will be amended to the CI document.
* **Audits and Reviews:** A review document will be produced, cataloguing the CM team’s  
  performance, noting any problems they are having conforming to the standard processes.

**Verification and Validation:**

Validation: Are we building the right product?  
Verification: Are we building the product right?

We can regularly inspect/test our code and design to spot defects in our program or logic and change/prevent them before they become an issue in the final product.

**Part 5: Work elements, Schedule, Budget**

-Activity Oriented Structure

* Planning
* Establish requirements
* Perform perquisite research
* Create work plan/Schedule
* Create budget
* Assign tasks
* Design
* Create written diagrams of starting ideas
* Design database
* Design website structure
* Create logical design for the software
* Agree on most optimal designs
* Ensure all requirements are satisfied by the design
* Development
* Create physical database
* Create working interface for prototype
* Combine developed elements into a rough prototype
* Evolve prototype into working version
* Testing
* Develop test procedures and checklist
* Perform tests on prototype
* Revise prototype based on test results
* Perform user testing and receive feedback
* Take feedback into account and make revisions
* Show final test version to client and receive feedback
* Take feedback into account and make revisions
* Release

- Release final product

Dependencies

* Project will depend upon a few programming languages which are known to our team of developers. HTML, CSS, JavaScript, Java, SQL
* POC will be the communication link between the team and the client.
* Test and trial errors, website will be updated with code and ran to test each and every part of the website
* Will provide technical overview in front of the client and a small audience to prepare for changes
* Final prototype to be released in May 2017 in front of client and audience. Final Project to appease client.