

## Contents

<b>1</b>	<b>2nd Deliverable Update</b>	<b>1</b>
<b>2</b>	<b>Project Description</b>	<b>1</b>
2.1	Critical Features . . . . .	1
2.2	Lower-priority Features . . . . .	2
2.3	Critical Frameworks . . . . .	2
2.4	Optional Frameworks . . . . .	2
2.5	Timeline . . . . .	3
2.6	Current App Progress . . . . .	3
2.7	Current Server Progress . . . . .	3
2.8	Implementation Notes: . . . . .	4

## 1 2nd Deliverable Update

1. The currently completed features are at the bottom of this file under the "Current App/Server Progress" sections
2. I have been pretty busy lately (with Senior Design, other classes, and whatnot), so my progress was slower than what was originally expected, but I finished a lot of the server requirements already.
3. Currently, on the application side, I have many of the Storyboard Screens prototyped already, so I just have to code the logic.
4. I have updated the timeline below to hopefully more accurately portray the expected completion of this project.

## 2 Project Description

I plan to make a scheduling service application similar to doodle.com. The following sections will describe the features and how I plan to implement the application on iOS.

### 2.1 Critical Features

- Allow users to register/login and persist their session using JWT (JSON Web Tokens)

- Allow users to copy/share a shareable key to join the meeting scheduler "room".
- Allow users to see a calendar with timeslots that people are available on.
- Will notify participants if all of the people who were invited all submitted their availabilities.
- Allow all participants to vote on a meeting time if there is are multiple.
- Will allow the winning timeslot to be added to the calendar

## 2.2 Lower-priority Features

I will choose to do one or more of these features.

- Users can choose/vote on a map for meeting place.
- In-room chat for notifying participants

## 2.3 Critical Frameworks

1. **AlamoFire**: for making network requests to a server.
2. **EventKit**: For accessing the calendar to add the event
3. **Starscream**: Websocket Library. Send room-specific information to server

## 2.4 Optional Frameworks

I'll be using one or more of these depending on what feature I plan to implement.

1. **MapKit**: For displaying possible meeting locations (May not use this)
2. **MessageKit**: For the chat (May not use this)

## 2.5 Timeline

Week	To do
Feb 25	Idea accepted. Get initial server with test routes running.
Mar 04	Continue working on API
Mar 11	Draft application UI
Mar 18	Start working on application
Mar 25	Login/Registration functionality finished
Apr 01	Working on Room Functionality
Apr 08	Working on Connecting app to backend
Apr 15	Work on optional additional features
Apr 22	Finish up application
Apr 29	Finishing touches to application

## 2.6 Current App Progress

Feature	Complete?
User Authentication	
User Registration	
Screen for displaying all rooms a user is a part of	
Create a new room	
Add other users to a room	
Common Calendar with All availabilities	
Add user availabilities	
Notify all users when everyone picked their availabilities	
Vote on best meeting time if there are multiple	
Export winning meeting time to apple calendar	
Calendar Component	yes

## 2.7 Current Server Progress

Feature	Complete?
User Authentication	yes
User Registration	yes
Create new room	yes
Add user to room	yes
Notify all members of room when everyone filled out their availabilities	
Protected Routes with JWT	yes

## **2.8 Implementation Notes:**

1. The unique key to join a room can simply just be the rooms id. This is probably not the best, but in our case, this should suffice.