Interaction Specifications
for
"Capital Games"

Team 2:

Jeff Adler
Eric Cuiffo
Nick Palumbo
Jeff Rabinowitz
Val Red
Dario Rethage

March 3, 2013
Version: 1

Contents

1 System Interaction Diagrams

1.1 Financial Data Retrieval Subsystem

1.2 Asynchronous Processing Subsystem

2   Plan of Work

2.1   Development and Report Milestones

Illustrated on the next page is a gantt chart reflecting our goals relative to the project deadlines. It incorporates both core development and report items. For our initial stages we focus on environment and platform set-up (i.e. deploying a development webserver) and the initial, core code implementation. At the same time we will finalize the details of our final product via the report milestones.

Development milestones have been spread out following the completion of the first report on 22 February 2013, beginning with deploying our development environment and server through Heroku from which we continue to our next milestone of deploying Ruby on Rails as well as all the Gems and API packages we are incorporating into our project, most notably Yahoo! Finance.

Report milestones are also set concurrently. As we begin to initialize our development environment, we will also build on top of and expand on previous reports to expand upon and fully realize the details of *C*apital Games.

Core goals leading up to Demo 1 include establishing all core functionality for *C*apital Games. This includes the following:

- **Rails framework-deployed core functionality :** This includes a working system for navigating the website, registering a new user account, logging in, and creating as well as participating in leagues.

- **Setting a foundation for the database:** On top of having the aforementioned core functionalities, they also must be able to pass data through a routed database.

- **Implementing the Yahoo! Finance API:**

- **A functional user interface:** Our website should be usable, and having a functional user interface from the start will give us a lot of room to expand and optimize the UI.

2.2   Breakdown of Responsibilities Introduciton

Contributions leading up to the completion of this report are covered in the "Contributions" table on the page following the gantt chart. For the future division of labor, we all plan on subdividing aspects of both the next reports as well as the development of the *C*apital Games Alpha.

2.3   Breakdown of Responsibilities

Responsibilities for server/development environment deployment and set-up will be shared between Val and both Jeffs, as all three equally have great experience in the subject. Meanwhile Nick, and Eric will work of sequence diagrams.

While all other diagrams on the report will be covered by both Jeffs, the User Interface Design and Implementation will be worked on by Val, Dario, and Eric. Nick and Dario will work on both data types and operations while Val and Eric also work on the traceability matrix.

System architecture and design will be covered by Nick and Val. Jeff R. and Dario will begin on the database structure and site routing via the Rails framework. Nick and Jeff A. will work on the implementation of users in the meantime.

From there we will further subdivide work on the final aspects of the website, likely sticking with our initial idea of splitting predominant responsibilities following the model, view, and controller (MVC) design pattern. Jeff R. and Jeff A. will lead work relating to model design, Dario and Eric will lead work relating to views and interface, and Nick and Val will lead work involving controllers. That being said, while the MVC pattern will model sub-component ownership among the team. Individual implementation responsibilities will be distributed a bit more evenly based on the particular strengths of team members.

In summary, Project Ownership will be based on the MVC architecture. To reiterate, Jeff R. and Jeff A. will have ownership over the Model portion, Dario and Eric will have ownership over the Views (user interface, etc.) portion, and Nick and Val will have ownership over the controllers portion. Even beyond Project Ownership, however, responsibility for the whole project will be shared and the success of our MVC architecture requires close coordination between all aspects.

Overall project success will be decided with how well the MVC component teams communicate and work with each other, as Capital Games will rely on the interactivity between the Model, Views, and Controller portions of the architecture.

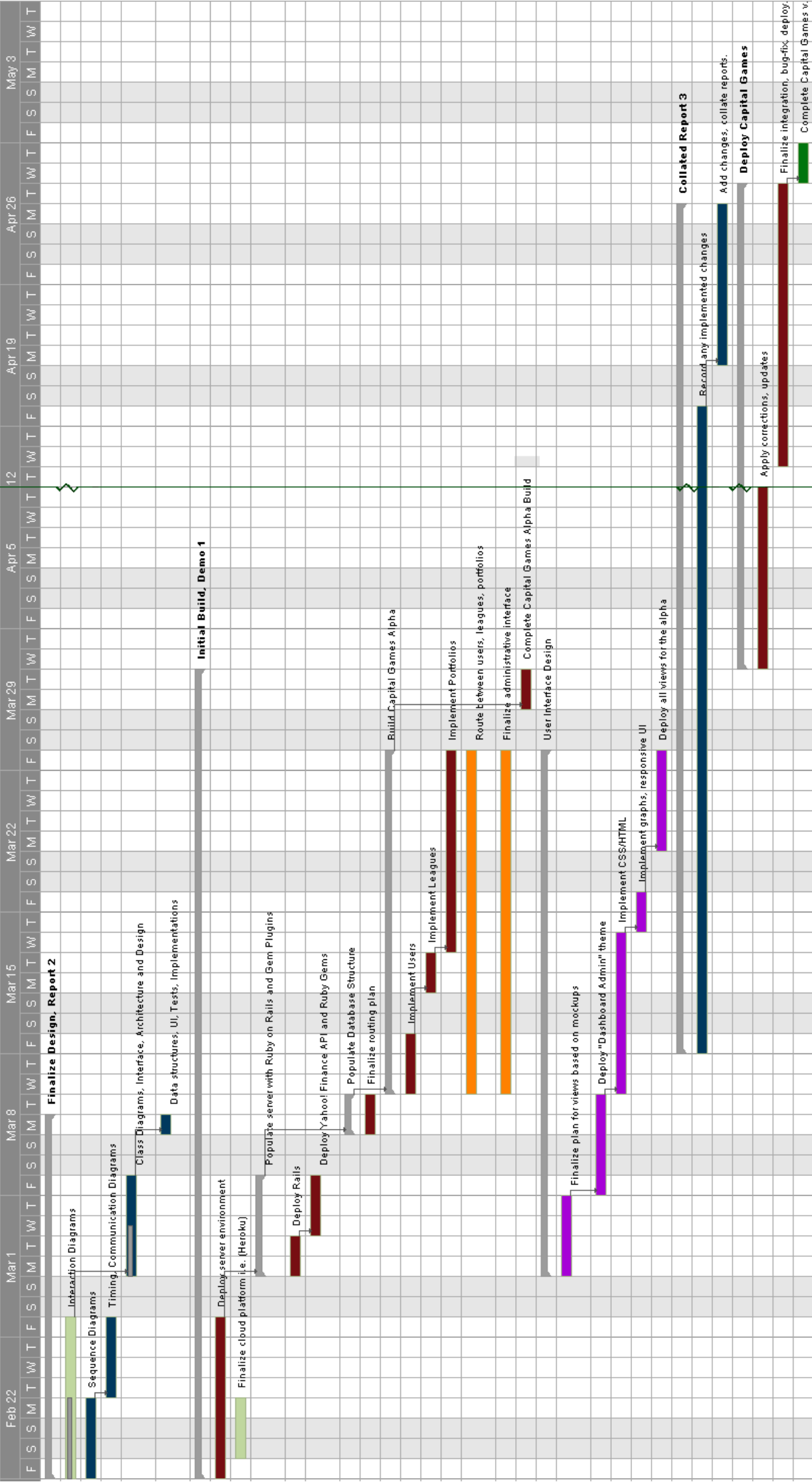## 2.4 Gantt Chart of Projected Milestones

Best viewed at 100% or greater:

Figure 2.1: This gantt chart projects how we will concurrently work on the project. All blue items are report-related, red and orange relate to the core project development and purple illustrates UI milestones.

3 Project Management

3.1 Coordination and Division of Labor

3.2 User Story Matrix

3.3 Report 1 Contributions

Category	Points	Names					
		Jeff A	Eric C	Nick P	Jeff R	Val R	Dario R
Project Management	10 Points	0%	0%	40%	50%	0%	10%
Customer Requirements	9 Points	20%	0%	0%	80%	0%	0%
System Requirements	6 Points	0%	33%	0%	0%	33%	33%
Functional Requirements	30 Points	30%	10%	35%	0%	5%	20%
User Interface Specifications	15 Points	0%	33%	0%	0%	33%	33%
Domain Analysis	25 Points	25%	25%	10%	20%	20%	0%
Plan of Work	5 Points	0%	0%	0%	0%	50%	50%