1 Overview

1.1 Purpose and Scope

Our team is interested in creating a website that allows users and food fans to communicate with each other by sharing recipes and rating them.

Recip.me distinguishes itself form other food web-services through the openness and the wide choice that it offers product-wise.

The project is going to be expanded in order to welcome food providers and producers

1.2 Goals and Objectives

The main goals of this application will be to develop a recipe database that will provide the following:

1. Collect recipes that the users will upload.
2. Allow users to submit feedback and change the recipes.
3. Allow users to order the products directly from the website.
4. Allow users to create a personal customed account.
5. Allow user to change temporarily the ingredients of a recipe while performing the order.

1.3 Project Deliverables

Date	Deliverable
09/23/10	Requirements Specification
09/28/10	Project Plan
10/05/10	Iteration #1 Plan
10/07/10	Technical Prototype
10/14/10	Customer Approved UI Prototype
10/21/10	Architecture Document
10/28/10	Iteration #1 Complete
11/11/10	Test Report
11/18/10	Iteration #2 Complete
12/02/10	User Guide and System Administration Manual
12/07/10	Product Released

1.4 Assumptions and Constraints

1.4.1 Assumptions

1. UMKC terminals are available and function when we need them.
2. Remote MAC Lab will allow our programmers to work away from UMKC.
3. Interfacing the app with a proxy server and third-party website is feasible.

1.4.2 Constraints

The following constraints have been identified for this project:

- The application needs to be written in HTML, CSS, PHP and SQL

1.5 Schedule and Budget Summary

1.5.1 Cost Estimate

1 project manager at 4 hours per week for 12 weeks 48 hours

1 requirements engineer at 4 hours per week for 12 weeks 48 hours

8 software engineers at 4 hours per week each for 14 weeks 448 hours

544 hours total

1.5.2 Schedule Summary

Step	Iteration	1	2	3										
Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
End Date	01-15	01-31	02-15	02-29	03-15	03-31	04-15	04-30	05-10	05-21	05-31	06-12	06-25	06-31
Design Architecture														
Specify/Evaluate Requirements														
Iteration 1 Development														
Iteration 1 Analysis														
Iteration 2 Design														
Iteration 2 Development														
Iteration 2 Analysis														
Iteration 3 Design														
Iteration 3 Development														
Iteration 3 Analysis														

1.6 Success Criteria

A working prototype, which is easy to use, that allows users to upload and buy multiple recipes.

1.7 Definitions

Term	Definition
Actor	user or other software system that receives value from a use case.
Project	Activities that will lead to the production of the product described here. Project issues are described in a separate project plan
Project	activities that will lead to the production of the Roo Balance application.
Scenario	one path through a user case
Product	What is being described here: the software recip.me website
Role	category of users that share similar characteristics
Use case	describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

2 Startup Plan

2.1 Team Organization

Role	Actor(s)	Responsibility

Role	Actor(s)	Responsibility
Project Manager	Adami	Call team meetings, coordinate communications within group, coordinate communications outside group, break out tasks, assign them to teammates
Developer	Lucchese, Segala, Rossignolo, Piccoli, Magrinelli, Adami	Develop software based on requirement and architect specifications
Programmer	El Amry, Bothalage,	Program to requirement and architect specifications
Tester	El Amry, Bothalage, Magrinelli, Segala, Piccoli	Write test cases, perform unit testing of test cases against incremental release of code, perform integrated testing of test cases against incremental release of code, report issues
Architect	Adami, Lucchese, Rossignolo	Specify overall internal workings of application
Requirement Engineer	Adami, Lucchese Rossignolo	Outline and document project dependencies and requirements. This includes internal and external dependencies.

2.2 Project Communications

Event	Information	Audience	Format	Frequency
Team Meeting	Task status: completed since last meeting & planned for next; obstacles encountered; change requests in process	All team members	Informal meetings following class; Formal meetings as needed; E-mail status updates & problems as they occur	As needed
Project Status Report	Review finished items, status of prototype; review any problems, schedule slippage, programming issues	All team members, customer	E-mail with information or In-person as customer sees fit	Iteration Closeout

2.3 Technical Process

An iterative and incremental development process is planned. Feedback will be used from each iteration to improve the next. The first iteration will focus on basic functionality of the application. Subsequent iterations will build upon that and incorporate more features as time allows.

2.4 Tools

· Programming & Markup Languages – HTML, SQL, JavaScript, CSS

· Operating System – iOS

· Version Control – all work products will be stored in a GitHub repository

· Development Tools – Atom, Sql Server Management

3 Work Plan

3.1 Resource Estimate

Detailed resource estimates are available in the linked file Team1 Estimated Effort.

In this document, tasks, roles, owners, and effort estimates & actuals are listed.

3.2 Release Plan

3.2.1 Plan By Feature

Iteration #1

Summary: First scratch of the site, allowing users-log in

Features / Deliverables	Estimated Effort	Actual Effort
Architecture / framework design	20	

Features / Deliverables	Estimated Effort	Actual Effort
Authentication / Credential management	50	

Iteration #2

Summary: Serving users with personal pages, using cookies

Features / Deliverables	Estimated Effort	Actual Effort
Personal Pages	70	

Iteration #3

Summary: Allowing users to modify ingredients of a recipe based on their needs, cooperating with food providers

Features / Deliverables	Estimated Effort	Actual Effort
Granting access to the site to food providers	50	

4 Control Plan

4.1 Monitoring and Control

The following list of dates includes formal reviews outside of the Communication Plan. Milestones are included to reference where the project is scheduled to stand as these reviews occur:

Date	Review / Milestone
08/02/2020	*Milestone: Technical Prototype Complete*
08/02/2020	5-Minute Status Report
14/02/2020	Manager's Briefing
30/04/2020	*Milestone: Iteration #1 Complete*
15/05/2020	*Milestone: Test Report Complete*
16/05/2020	Inspection
21/05/2020	*Milestone: Iteration #2 Complete*
31/06/2020	*Milestone: Product Released*
15/07/2020	Final Presentations

4.2 Configuration Management Plan

The following procedure is to be used when making changes to all baselined work products:

1. All project work products will be stored in decentralized GitHub repository.
2. All baselined documents will have a Document Control section with a change history to track initialization and subsequent changes.
3. All project work products (documents, source code, test cases, program data, test data, etc) will be stored in the GitHub repository but not all will be under change control (subject to formal change control procedures.) Only the system requirements, project plan and source code will be baselined and under configuration control.
4. Items that are subject to change control will be considered baselined after a group review at the end of the initial document creation.
5. The change control procedure once a product is baselined is:

(1) anyone wanting to make a change to a baselined item has to let the rest of the team and project sponsor know, describing the change, reason for the change, expected schedule impact, and time line for integrating the change.

(2) If no reason is found why change request shouldn't be permitted, it will be considered accepted and the person proposing the change may proceed with the change.

(3) if anyone does object to the change, the reason for objecting will be discussed at a meeting where everyone is invited to attend and voice their opinion. At the end

of the meeting a democratic vote will be held to decide whether or not the change should be allowed.

(4) if a change takes place, the initiator must collaborate with the project manager to update the schedule

5 Supporting Process Plans

5.1 Risk Management Plan

Rank	Risk	Probability of Loss	Size of Loss	Risk Exposure	Response
1	Schedule / time line delivery	Likely	Major	High	Mitigate: Stick to the schedule.
2	Learning curve for new tools and technologies longer than expected	Likely	Moderate	Moderate	Buy Information: Begin working on a basic prototype early to test out fundamental programming concepts & knowledge

5.2 Test Plan

The test plan defines the items that will be tested, methods for testing, and a schedule detailing the tasks, owners, and time line.

5.3 Product Acceptance Plan

At the conclusion of each iteration, the prototype created will be tested to ensure it meets the requirements of that iteration. For testing out the product changes, the test product will be hosted by altervista, with another url.

For the final iteration, product acceptance testing will ensure that the prototype functions as expected with a user's data.