RMIT UNIVERSITY

Project MatchMe
PROJECT CHARTER

Version: V1.0
Date: 20/03/2017
Sponsor: Amir Homayoon Ashrafzadeh of RMIT University
Number: Project #1
Author: Jasmine Ellis
Hao Li

Haotian Xu
Xieyang Wu

Commercial - in – Confidence

Document Control

File Directory	https://drive.google.com/open?id=1c1BEaq1NZb9eZFFdlUWPsjp-X7Bkh_y9NarP-pKLayI

Distribution

Version	Issued	Recipient	Entity / Position
V 1	20/03/2017	Amir Homayoon Ashrafzadeh	Product owner and sponsor
V 2	13/05/2017	Amir Homayoon Ashrafzadeh	Product owner and sponsor

Amendment History

Section	Page	Version	Comment
Project Approach and Methodology	6	1	Section was expanded upon and more details were added.
Project Scope & Deliverables	8	1	A list of what is considered “In Scope” was added.
Project Summary	5	2	Modified description of matching function, which has changed from Q&As to hobbies / interests.
Project Methodology and Approach	6	2	Updated what we use for deployment from AWS to Heroku. Changed MongoDB to PostgreSQL.
Project Scope & Deliverables	9	2	Added admin functions to scope. Modified matches index section to better represent deliverable.
Project Scope & Deliverables	10 - 12	2	Updated “Deliverables” section to better reflect our schedule.

Staff or Entities Consulted

Name	Position / Organization
Amir Homayoon	Product Owner / RMIT University

Ashrafzadeh	

Related Documents

Name	Author	Description
N/A	N/A	N/A

Preface

The purpose of this document is to outline the Charter for MatchMe. It serves as an agreement between the project team, the sponsor and the supervisor. It outlines the project's purpose and how the project will be approached, resourced, managed and delivered. Any amendments after this document has been signed off will be via addenda.

Table of Contents

1 Project Summary

MatchMe is a online dating web application that features a matchmaking system, in which users can find their “matches” based on their hobbies / interests, as well as other factors such as age, gender, and location. The matches index will be ordered by most to least likely to be a match to the user. Users will also have the option to “Like” other users. Functions will include login, logout, register, delete account, view matches, like/unlike a user, view likes, view users who like you.

2 Project Sponsor

The project sponsor is Amir Homayoon Ashrafzadeh of RMIT. RMIT is a public University based in Melbourne, Australia. It is ranked as a five star university by QS World University Rankings. The CSIT department of RMIT provides a number of internal projects each year for it’s students to gain practical experience.

3 Stakeholders and End Users

The key stakeholders are RMIT University (organisation), Amir Homayoon Ashrafzadeh (Project sponsor, product owner, and end user), Jasmine Ellis (team member), Hao Li (team member), Haotian Xu (team member), Xieyang Wu (team member). As project sponsor and product owner, Amir oversees the project, prioritises the items in the backlog, signs off on deliverables, and sets the requirements. He (in theory) has a stake in project’s ability to produce a profit. Jasmine, Hao, Haotian, and Xieyang are responsible for delivering the product. This includes project planning, design, documentation, testing, and implementation.

4 Appointment of Project Leader

The project leader is Jasmine Ellis. The project leader was appointed because she has the most experience with the Ruby on Rails framework and the MVC design pattern.

5 Project Team Members

The project team members and their respective roles are:

- Jasmine Ellis
    - Project leader, scrum master, developer, tester
- Xieyang Wu
    - Developer, tester
- Haotian Xu
    - UI/UX designer, developer, cloud engineer, system administrator
- Hao Li
    - Developer, cloud engineer, database engineer, risk manager

6 Project Methodology and Approach

Location

All meetings will be conducted at RMIT city campus. Monday meetings will be conducted in various meeting spaces around campus. Wednesday meetings will be held in 057.03.002.

Methodology framework

The process followed will be SCRUM. This methodology was chosen because we want to ensure that our project is readily adaptable to any possible changes in requirements. We also chose this methodology so that we can get immediate and continuous feedback from the client.

Each Sprint will be one week long. All deliverables are to be signed off by Amir before being considered “done”. Another Sprint cannot be initiated until the one before it is complete.

Weekly Deliverables:

- Implementation of one or more functions (depending on their size). This will involve processes related to
    - Planning and design
    - Implementation (construction and coding)

    - Testing and debugging
    -
- All documentation relating to the next planned Sprint
    - User stories
    - Wireframes
    - Test cases

Project Tools

- Trello - For keeping track of all project tasks
- Slack - Main communication method between team members.
- WeChat - Communication method between team and product owner
- Cloud9 - Project IDE.
- Github - Code repository
- Heroku - For project deployment
- AWS S3 - For user image storage
- Google Docs - For generating and sharing documents

Technology Stack

- Ruby on Rails
- PostgreSQL
- HTML5
- CSS3 and Sass
- Bootstrap 3
- Javascript

7 Project Governance

The Governance model is as follows:

COMMUNICATION

Communication between the team and the project sponsor / owner will take place via email, as well as during weekly meetings. The team will communicate amongst themselves via email, Slack and bi-weekly meetings. Backlog and Sprint tasks will be communicated amongst team members and product owner via Trello. Documents will be shared both

on Trello and Google Drive.

BI-WEEKLY MEETINGS:

1. **Standup**
    a. When: Monday from 3PM to 5PM
    b. Where: RMIT City Campus
    c. Attended by: team members
    d. Discussion recorded via Meeting Minutes document.
    e. Discussion topics include:
        i. What have I done since the last Scrum meeting?
        ii. What will I do before the next Scrum meeting?
        iii. What prevents me from performing my work as well as possible?
2. **Sprint Review, followed by Sprint Retrospective and Sprint Planning**
    a. When: Wednesday from 8:30AM to 10:30AM
    b. Where: RMIT City Campus (057.03.002)
    c. Attended by: team members, product owner
    d. Discussion recorded via Meeting Minutes document by Jasmine Ellis.
    e. Discussion topics include:
        i. Sprint Review: potentially shippable product demo.
        ii. Sprint Retrospective:
            1. What should we stop doing?
            2. What should we start doing?
            3. What should we continue doing?
        iii. Sprint Planning: which stories will be done during next Sprint. Trello is updated by Jasmine Ellis to reflect decisions made during this phase of the meeting.

CHANGE CONTROL:

We will prevent scope creep by setting project scope and ensuring everyone understands what we will be doing and what we will not be doing, at least from a high level perspective. We will define our high-level deliverables and have them approved by the product owner.

Changes to the project will be considered out of scope if they:

- Involve re-doing an already completed function.
- Involve a significant amount of work that would, at the judgement of team members, result in the project failing to be completed by the deadline.

Risk Management

A Risk Register will be completed by the team to identify potential problems and formulate strategies to deal with them should they arise. Furthermore, team members will have an opportunity to voice any problems or impediments to their work during the Sprint Retrospective.

Risk response will include:

- Mitigation
    - Core Tasks and Stretch Goals will be clearly defined. Team focuses on Core Tasks and attempts Stretch Goals only after Core Tasks are complete without the occurrence of potential risks..
- Avoidance
    - In which the Product owner removes a requirement to avoid any risks associated with it.
- Containment
    - In which risks are dealt with as they occur. Responsible parties are identified and a strategy for minimising the effects of the risk are planned and implemented as needed.
- Evasion
    - In which low-impact risks are identified but no action is taken as their occurrence would not impair the project.

8 Project Scope & Deliverables

All deliverables will be signed off by Amir Homayoon Ashrafzadeh during the weekly Sprint Review meeting.

In Scope:

- Basic user functions
    - Homepage
    - Login/logout
    - Registration
    - Upload profile image
    - User profile
    - Edit profile
    - Account deletion
- Admin functionality
    - Admin can view all users on Users Index page
    - Admin can delete users
- View matches index
    - All matches generated for user viewable on Matches page, which is paginated
    - Matches ordered by match percent
    - User can click on match to view their profile
- Likes function
    - User can like other users by clicking a button on their profile.
    - Users cannot like themselves.
    - User can also view a page showing
        - Who they liked
        - Who has liked them

Out of Scope: Messaging system, uploading multiple images, password reset mail, account activation mailer, new like notification.

DELIVERABLES:

22/03/17

- Documentation:
    - Templates:
        - Homepage
        - Registration
        - Login
    - User Stories:
        - Login
        - Logout
        - Register
    - Login test cases:

        - Valid login
        - Invalid login
        - Login with remembering
        - Login without remembering
    - Registration test cases
        - Valid registration
        - Invalid registration

29/03/17

- Implementation:
    - Login / logout
    - Homepage
    - Registration
- Documentation:
    - Wireframes:
        - User profile
        - Edit profile
    - User stories:
        - View profile
        - Edit profile
        - Delete account
    - Test cases:
        - Valid edit profile
        - Invalid edit profile
        - Delete account

05/04/17

- Implementation:
    - View profile
    - Edit profile
    - Edit account
    - Admin functionality
        - Users index page
        - Delete other users
        - Cannot delete self
- Documentation
    - Wireframes:
        - Matches index
    - User stories:

        - View matches
    - Test cases:
        - View matches

12/04/17

- Implementation
    - User image upload
    - Profile modelling

26/04/17

- Implementation
    - CRUD operations for profile
    - Account settings

03/05/17

- Implementation:
    - Matching algorithm

10/05/17

- Implementation:
    - View matches index
        - Matches determined by matchmaking algorithm Input into the algorithm includes user preferences, profile data, and interests.
        - Match objects only generated between two users if they both match their gender preference and they are nearby each other (within 50 miles if “nearby” attribute is false, within 15 miles if “nearby” attribute is true”).
        - Users can click other users that appear on Matches index and view their profile.
- Documentation:
    - User stories
        - Like user
        - Unlike user
        - View likes
        - View liked by
    - Test cases
        - Like user
        - Unlike user

        - Like self
        - View likes
        - View liked by

17/05/17

- Implementation:
    - Like user
    - Unlike user
    - View likes
    - View liked by