*Macromatic*

*A user-friendly code-generation tool*

*For people who want a little more function,*

*Without the programing knowledge.*

Robert David Warner

Neumont University

Project Success

In the entertainment world, users gain a significant advantage by using in-game macros (sequences of game-instructions aliased to a single execution); however, writing these macros can be tedious – requiring knowledge of each specific game’s syntax. Macromatic is a web-application that will facilitate macro-generation (on first iteration only supporting a single game, but supporting additional games in the future). Using my application, users will be able to create macros tailored to their specific needs by navigating though GUI-options on a single webpage. The application will transform a user’s needs into game-specific macro-syntax, provide a clipboard option to copy the macro (to be later pasted into the game directly), and potentially provide the option to publish the macro for other user’s to see and copy. I propose to deliver a functional open-source web application that is publically accessible and demonstrates the proof of concept for my design idea using World of Warcraft (WoW) as my first supported game.

Target Users

1. Electronic entertainment enthusiasts (specific to WoW) with little to no code experience.
2. Electronic entertainment enthusiasts (specific to WoW) with any level of enthusiasm **but little time**

*\* More target users to be added in future iterations*

Project Backlog

----------------------------------------------------------------------------------------------------

Backlog Item Naming convention: (Project-Acronym)(Item-Type)(Unique-Number)  
 ex: M2S10  
 Item Type Key:

* S : User Stories – representing new functionality
* B : Bugs – representing work to address a defect
* C : Chores – representing work that must be done, but provides no direct business value
* P : Prototypes (a form of risk mitigation) – representing proof of concepts that help provide information for decision making around whether some functionality may be valuable.
* E : Epics – representing big user stories. (needs to break them up)

**1.)** Title**: M2E1** – WoW Macro Generation Functionality

Description**:** WoW users from target group ***a*** used ***b*** will benefit from the ability to auto-generate macros for abilities and activated-items based on relevant applicable game-preferences

Acceptance Criteria:

* WoW syntax macro is produced from user selections
* Client-side validation exists for WoW relevant user selections
  + Users are protected from their own stupidity

------------------------------------------------------------------------------------------------

**2.)** Title**: M2S1.1** – WoW Macro Generation Functionality - **GUI**

Description**:** WoW users from target group ***a*** will benefit from a comprehensive graphical user interface wizard that leads them through the entire macro creation process

Acceptance Criteria:

* GUI does not suck for users (the app will not do anything without the user’s permission; and will present clear choices to the user for what they want to build).
  + Think of what you would have wanted out of healthCare.gov’s GUI wizard

------------------------------------------------------------------------------------------------

**3.)** Title**: M2S1.2** – WoW Macro Generation Functionality – **Efficiency**

Description**:** WoW users from target group **b** will benefit from the macro creation process being quick.

Consideration of anything that facilitates a faster macro- creation process.

Ex:

* No login required for use of service
* Single Page application

Acceptance Criteria:

* + Macro Creation process is all done on a single page.
  + Macro creation requires no user authentication (\*see **M2P1**\*)

-------------------------------------------------------------------------------------------------

**3.)** Title**: M2P1** – User created Macros are optionally publicly viewable / hosted

Description**:** Explore the potential for hosting pre-created user macros that have been marked as ‘public’.

Acceptance Criteria:

* + Were this to be implemented, previously created ‘public’ macros will be persisted in some way
  + Were this to be implemented, view-page will exist for all previously created macro-names
  + Were this to be implemented, previously created public macros will their own view pages
    - These view pages will contain a description of the macro’s purpose, generated from the user selected preferences
    - These view pages will contain the same clipboard copy function as on the result of **M2E1**

----------------------------------------------------------------------------------------------------

Milestone Plan

**Sprint 1:**

**Title:** Project Creation + First Sprint Backlog task items (see project tracking board for specific tasks.

**Description:** Create GitHub repo and set up folder substructure and first sprint branch; create sprint backlog with tasks assigned to it. Create project tracking mechanism and create user add user stories / acceptance criteria.

Link Mr. Garlic to Repo and Project Tracking website.

**Acceptance Criteria:**

* GitHug repo exists for project.
  + Professor has access
* Trello/Youtrack board exists for project
  + Professor has access
* All Sprint Backlog items are completed.

---------------------------------------------------------------------------------------

**Sprint 2 & 3:** Subject to change based on future Sprint review and Sprint backlog