**CS 408 Individual Project Scope and Outline Plan – Duologico**

Proposal

Duolingo is a phone app, almost a game, which allows learning a second language to be surprisingly accessible. The proposal of the project is to create an app, similar to Duolingo, that instead teaches the user about propositional logic and certain programming languages. The app will attempt to integrate many of the relevant features used by Duolingo such as repetitive based learning, periodic tests and gamification of the user’s progress, while stripping those that are not appropriate.

A major part of Duolingo’s appeal and success is its gamification. This process involves taking somewhat mundane tasks and packaging them in a way to keep the user infused and eager to come back. Some key examples of this process are “levelling” systems: the player can only progress if they get 100% in a test; or in game rewards if the player uses the app multiple days in a row. Leaderboards for test results is another great example of gamification and user may be encouraged to compete against friends. In Duologico, these systems will be a major part of overarching design.

Key Features

* A variety of multiple choice quiz questions
* A variety of text entry quiz questions
* Gamified progress tracker ie level up system
* Detailed information tabs about topics
* Leaderboards for timed quizzes
* Post progress to social media
* Notifications to encourage learning

Related Work

As the project is roughly based on Duolingo, a lot of inspiration will be taken from its design and functionality. Obviously, there are some major differences between learning formal and informal languages but many of the underlying systems should still be relevant to this course.

As for programing language specific apps there are a few to choose from. Notably, Udacity provides lectures and interactive quizzes for learning software development. While the lectures are clearly out of the scope of this project, research will done into how these quizzes differ from those on Duolingo and how the techniques used can benefit Duologico.

Methodology

Before any programming is started, a list of features and specific implementation features will be finalised. Once the features are decided, the design of their implementation will be worked on until the entire system has a solid design spec. Throughout implementation the design specification will be tweaked as required. If enough time is left at the end of development, bonus features may be considered.

Evaluation

The final product will be tested rigorously via the use of a javascript testing framework such as QUnit. The app itself will also be tested by testers who may fill out a short questionnaire about their experience with the app. Hopefully, this test data will be received early enough in development for it to be used to fix certain design issues with the app. The implementation will be cross referenced with the original design to ensure that the final product is as specified.

Rough Schedule

Week 1 – Key feature design, how the quizzes will work, levelling system etc.

Week 2,3,4 – Software Design

Week 5 – 8 – Implementation + necessary redesign

Week 9 - Testing, both programmatically and socially

Week 10, 11 – Report and hand in (report shall be worked on throughout but focused on here)

The Project will use the “Software Development-based Project” Marking Scheme.