University of Nevada, Reno
Department of Computer Science and Engineering

Project Title: <u>Guild</u>

Team #21
Riley Moore
Blake Cash
Ryan Van

Instructors:
Sergiu Dascalu
Devrin Lee

External Advisors:
Chase Carthen
Mitchell Martinez

March 15, 2019

Table of Contents

Project Progress Demo Presentation

The below is a copy of the handout given out at the demo, containing elements (1) & (2):

GUILD
Team #21
Blake Cash, Riley Moore, Ryan Van

Status of the Project

Guild's Framework is, largely, done.

Core and baseline functionalities like flask integration, communication with the database, posting and retrieving events, and some basic tools have all been implemented.

With the major exception of a login token functionality, and event registration (both of which which we intend to implement next), Guild works as a completely functional, although extremely barebones, gaming event planning application.

Progress Made

Below is a list of features, functionalities and updates we intend to show off during this demo. A formal list of use cases (I.E. what we showed on Project Part 2) is available at the back of this packet.

- Toolkit implementations (Following tools are all functionally complete, may have UI updated)
    - D&D Initiative Tracker
    - Super Smash Brothers Ultimate stage striker
        - Used at a real (100+ entrant) tournament!
    - D&D (and general tabletop) dice rollers
    - Lifepoint and health CCG tracker/displays
        - Magic the Gathering
        - Yu-Gi-Oh!
- Completely updated database schema
- Event posting and viewing functionality
    - Flask integration/data binding
- Bug Fixes (A lot!)
- Visual updates
    - Scene transitions
    - Screen-to-screen parity

What We Plan to Implement

Generally in order (So we'll tackle things at the top of the list first.) You'll notice everything on our formal use cases is still intended to be finished by the end of the semester.

- Event deletion
- Bracket creator
- Login token system
- Event registration
- Private events
- Event searching
- Push notifications
- User profiles
- SQLAlchemy

Formal Use Cases in this Demo - Where we're at.
(These are taken straight from project part 2)

ID	Use Case	Description	ETA
UC01	CreateEvent	User will be able to set the date, time, location, game type, and general information of their event. User/users who create an event will have admin permissions, which enable them to manage their event all the way from beginning to end. Admins can either send it to the database as a public event or as a private event which only certain users or groups can see.	COMPLETE
UC02	EventRegistration	Users will be able to register for an event listed on the applications and have it propagate onto their calendar and notifications. A user may register into an event in two ways: by accepting an invitation to either a public or private event, or by selecting a public event after having searched one out and registering.	Contingent on login
UC03	FilteredSearch	User will be able to select specialized categorizes to minimize the amount of items they are searching for. Filters for game type, location, and minimum host ratings will be available for the user to enable or disable when looking for an event. Users may search for both players and events through this system	CUT
UC04	UploadData	The user will be able to upload	CUT

		files necessary for the running of events and have them hosted on the database. Users may also upload their google calendar so that groups may coordinate events when users are most available.	
UC05	Calendar	User will be able to view all the events that they have registered for based on a month, week, day format. On the day format, the events will be listed by the hour it starts and ends.	CUT
UC06	Tools	User will have access to a variety of tools to assist with their tabletop and LAN gaming experience. (i.e dice roller, deck list, tournament brackets, etc.) At the main menu, a user will select the tools options which will navigate them into a subsection of the application with more specific features. For example, if a user wants to access a specific dice for Dungeons and Dragons, he will navigate through tools > tabletop > dice rollers > D20.	COMPLETE - Can add more with extra time
UC07	UpdateProfile	Users will be able to provide additional information regarding what their interest are related to the platform. (i.e favorite board game, prefer being Dungeon Master, etc.)	Integrated w/ google login
UC08	EventDeletion	User will be able to delete an event that they have created. Only the admins or the creator of the event will have the permissions to delete the event.	Contingent on login
UC09	MainMenu	The app will have three main options after logging in. The user may select Events, Profile, or	COMPLETE

		Tools, which will all bring the user to a new set commands specific to that selection.	
UC10	PushNotifications	If a user has listed themself for an event, a notification will alert the user a certain amount of time before to remind them of the event. The user can tap this notification to link them to the event page within the app. Push notifications may also be enabled to see when a user has been messaged.	???
UC11	GroupManagement	A user may navigate to the group creation menu from the main menu, invite other users to a private or public group. Users in the group may create and share events and also chat amongst each other.	By final demo
UC12	BracketManager	Admins will be able to create, edit, and post tournament brackets to the database, while regular users will be able to view the bracket and upcoming matches. Brackets will also keep track of character and deck choices and use this data to update user profile information.	Within next week
UC13	DnDAssistant	Using a predetermined group of players, the application will allow users to cycle through each player based on their given initiative score and display relevant stats like armor class and health.	Within next month

Current Project Status

Currently, the Guild application is at the point where the framework is mostly complete with a few exceptions, and the app is considered to have much more functional use than before. In terms of progress made since the December demo, the database connection from the application is completely finished, which was an important milestone for the team. The UI skeleton is primarily done, but may still be subject to aesthetic and layout changes. In addition to framework progress, the team has completed a lot of work on the functional side, including many of the tools the team wanted to include like dice rollers, an initiative tracker, a stage striker, and card game calculators. Using flask, the database is successfully taking input from the app to create events in the AWS database, and it is relaying those events to applications which connect to the service. The database itself has also been reworked significantly since the December demo.

As far as work that still needs to be done, the team still needs to finish the last main framework component, that being the onboard SQLite database. The SQLite database is crucial for each user to store relevant information on events they have signed up for, and for the completion of the in app calendar. In addition to that, the team needs to finalize the event management side of the application, which includes event searching and registration. Data binding implementation will be needed to make sure that the application is populated with events as they are searched or registered for. Finally, on the tools side of the app, the bracket generator still needs to be completed as it is a crucial component to many LAN gaming events.

Team Contribution

Blake Cash: Blake, in addition to preparing the handout and slides presented during the demo, put most of his actual code work into the tools that are seen on the app itself, such as the initiative tracker and dice rollers. Blake, like all other members of the team, spent time on flask integration and database-to-phone communication. In total, Blake spent approximately 12 ½ hours on this project part.

Riley Moore: For project part 4 Riley focused his efforts on completing the database to application connection with Ryan and Blake in order to have it in a finished and presentable state. Riley wrote the code pertaining to all menu management, as well as event creation/deletion. Riley spent approximately 12 hours on this project part.

Ryan Van: For project part 4, Ryan focused on assisting Riley and Blake to get the front-end of Guild to connect to the database. Ryan worked to on the Flask code to allow for Guild to add events to the MySQL database hosted on AWS. Ryan also

created the new database that should be the finalized version used for Guild, unless new applications are added to Guild. Ryan spent around 11 hours on this project part.