# LoLTalk

Your friend, and source, for gaming in the League of Legends

By: John Pick

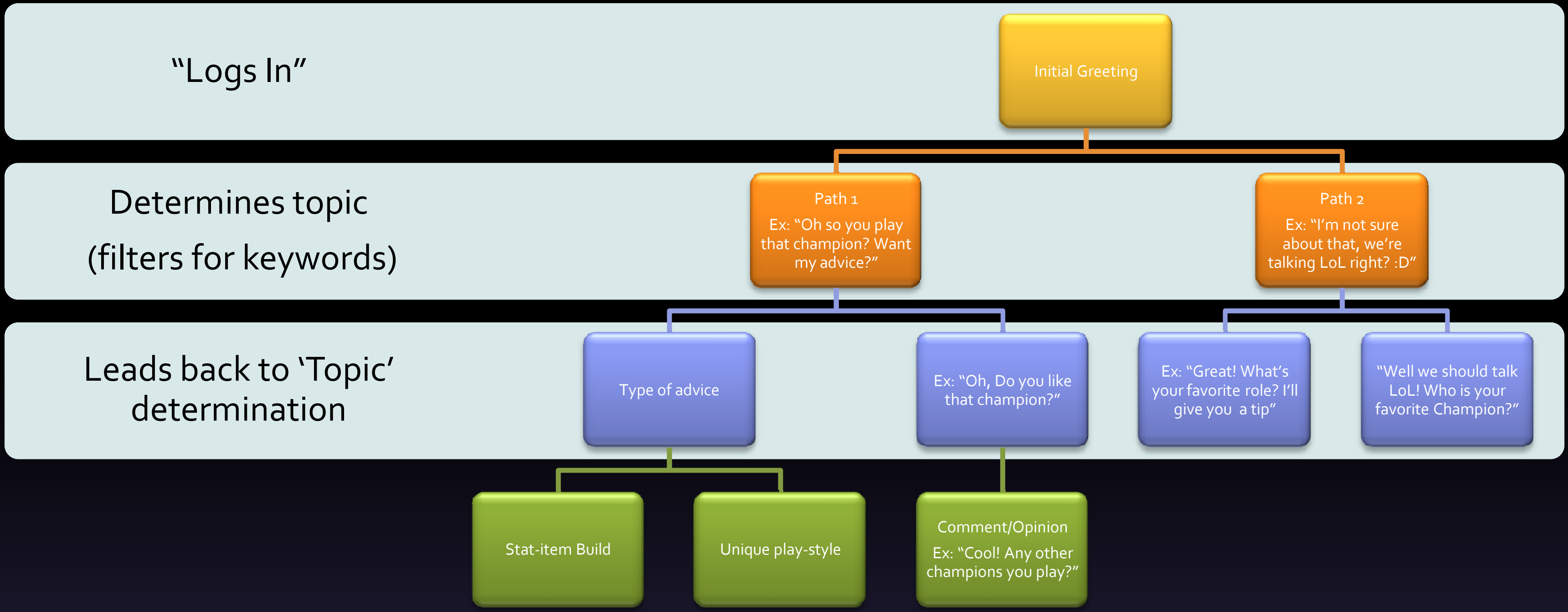
# Introduction

LoLTalk acts as:

- Your “nerdy” friend who plays the online computer game ‘League of Legends’ all day, everyday and only stops just to talk to you… about LoL. He assumes you are a LoL player and already knows the basics.

- LoLTalk loves to give advice.
  - Meant to act as a guide source to playing the game. LoLTalk will attempt to drive the conversation down a path into helping you (his friend) become a better LoL player
    - Gives tips on how to play an individual character
    - Gives tips on how to succeed in an individual role on your team (takes into account that certain characters can be played in different roles)

- LoLTalk has his own opinions…
  - Every player has things he or she loves and hates about LoL and LoLTalk is no exception
    - If LoLTalk does not end up giving advice on the game, he will attempt to “talk LoL” with you… he has plenty to say on the subject.
    - LoLTalk is bias and “voices” them depending on what the user says.
      - His bias also influences the kind of advice he gives



"Logs In"

Initial Greeting

Determines topic

(filters for keywords)

Path 1

Ex: "Oh so you play that champion? Want my advice?"

Path 2

Ex: "I'm not sure about that, we're talking LoL right? :D"

Leads back to 'Topic' determination

Type of advice

Ex: "Oh, Do you like that champion?"

Ex: "Great! What's your favorite role? I'll give you a tip"

"Well we should talk LoL! Who is your favorite Champion?"

Stat-item Build

Unique play-style

Comment/Opinion

Ex: "Cool! Any other champions you play?"

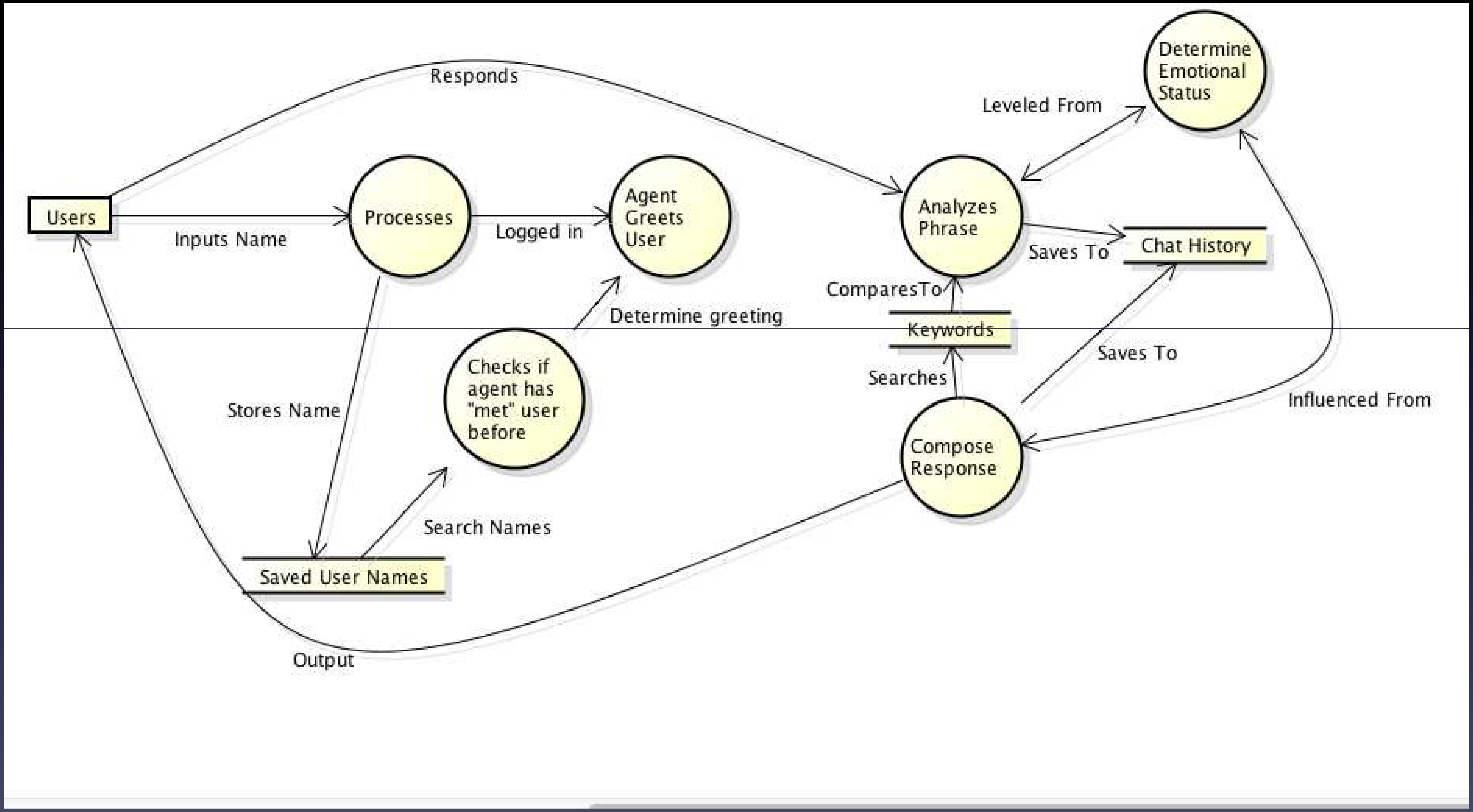
- Keywords are categorized (such as by character names, roles, yes/no answers, and greetings) to determine which path to call on. Keyword categories are checked depending what level of the graph the user is on
- All paths would eventually lead back to the 2nd row on the graph or until a "goodbye" phrase is sad

# Happiness Meter (to be made)

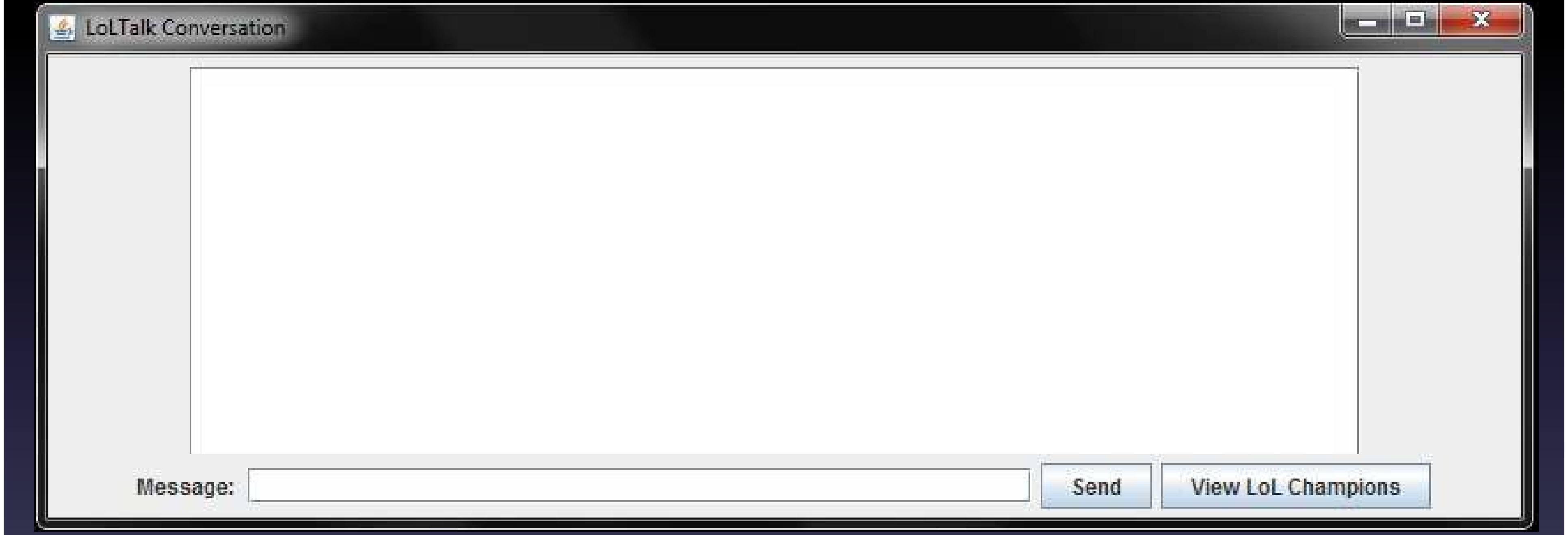
- Like most of the LoL community… LoLTalk may start off happy but depending on what you say, you might find him upset with you. For example
    - Not staying on the topic of League of Legends
    - Disagreeing with him (ex: “that’s bad advice”)
    - Claiming to like a character he hates

- Each time you say something that upsets LoLTalk, his meter will move towards an “upset” point. As he reaches his responses will be less enthusiastic. Reaching the “upset” point will trigger a different path from the others where he makes it clear he’s upset with you.
  - He even may or may not decide to help you with his LoL advice until you make him happy again.

# DFD Model


Responds
Determine Emotional Status
Leveled From
Users
Inputs Name
Processes
Logged in
Agent Greets User
Analyzes Phrase
Saves To
Chat History
ComparesTo
Determine greeting
Keywords
Saves To
Searches
Stores Name
Checks if agent has "met" user before
Influenced From
Compose Response
Search Names
Saved User Names
Output

# GUI



# Sample conversations

- LoLTalk: Hello! Who is this?
- > John
- LoLTalk: Hi John! Nice to meet you!
- John: nice to meet you to
- LoLTalk: Thanks! I just got off a game of LoL :D we won of course
- John: cool
- LoLTalk: Which champion did you last play as?
- John: Draven
- LoLTalk: Oh so you play Draven! I LOVE DRAVEN! He can really only be played as an ADC. Need any advice on that Champion?
- John: sure
- LoLTalk: Well, Draven is an ADC so you'd simply build attack damage items. I would rush for either an infinity edge or a blood thirster first followed by a phantom dancer.

- LoLTalk: Hello! Who is this again?
- > John
- LoLTalk: Hi John! Nice to talk to you again! :D Have you been playing more LoL?
- John: No.
- LoLTalk: Oh, well that's too bad. But I have! Any champions you've been playing with?
- John: I've been using Ahri
- LoLTalk: So you play Ahri? That's cool. Need any advice on that Champion?
- John: No thanks
- LoLTalk: Oh, do you like that Champion?
- John: yes I do
- LoLTalk: Awesome. I like that Champion :D Any other champions you like to use?
- John: Can you give me advice on how to play Ahri?
- LoLTalk: Sure! Ahri is an AP Carry, that means you should rush magic damage. I would say to start off buying a couple doran rings before buying the more expensive viable magic items :P
- John: bye
- LoLTalk: goodbye!

- LoLTalk: Hi John! How was your LoL gaming been going?
- John: I have nothing to say
- LoLTalk: I'm not sure about that, we're talking about LoL right? :D
- John: yup!
- LoLTalk: Great! What's your favorite role? I'll give you a tip
- John: jungle
- LoLTalk: Oh I love juggling. For juggling you would need to use a character with high sustain and usually build it with a half defense half damage style in mind.
- John: thanks
- LoLTalk: No problem! Any other champions you've been playing with?

# Software Development Life Cycle

- Originally planned on simply doing the Waterfall Model
  - Seemed straightforward and more “my style”
  - I intended  to plan my system features, code it, test it and submit it.
  - …Learned pretty quickly it wasn’t going to work (project requirements weren’t really set in stone, kept on having to change my idea for features. When I actually  started what I did didn’t really even fit the Waterfall Model anymore)

- Started to switch to Rapid Application Development (RAD)
  - Worked because I was both the coder and the user, the “joint planning and developing” went quite smoothly.
  - Worked with short development times
  - Worked with the style of coding functions in “increments” (made each proper response work one after the other)
  - Planned new ideas for workable functions as current functions were being completed

# Challenging Aspects

- Coding…. Lots of work
  - Lots of debugging I had to (and still ongoing) do with each new path I tried to implement
- Time Constraints – Need to work on managing my time better
- Inputting all the data…

# Rewarding Aspects

- Felt like I learned more of and can appreciate the SDLC more so than before by having actually implemented parts of it myself
- Learned to start practicing better time management
- Coding was also fun to build a system around a game I play – It can be entertaining to make the system say what you want sometimes…
- Learned some new coding skills (how to build a GUI in java) and gained a new appreciation for my ability in coding “bigger” projects than just the labs

# Questions?