

AtlasBot

Contents

1.	AtlasBot Concept and Explanation	2
	What is AtlasBot:	2
	How will AtlasBot work?.....	2
	What data does Atlas/AtlasBot store?	3
	How will AtlasBot be build?.....	3
	Who will the AtlasBot Team be?	3
2.	Analysis.....	4
	Current state of Bots within the League of Legends community.....	4
	What will AtlasBot's selling feature be?.....	4
	User Support.....	4
	Usability	4
	Modularity	4
3.	AtlasBot Design.....	5
	Database.....	5

1. AtlasBot Concept and Explanation

What is AtlasBot:

AtlasBot started out as a simple small project for the Atlas LoL Matchmaking server. This bot could set roles according to someone's League of Legends account and match players up who meet the criteria. I soon afterwards realized the potential within this project and how it could be expanded to fit other servers their needs. AtlasBot expanded to a new height with the focus on making getting and assigning roles much easier.

Every champion mains server has some sort of role, it either is Ranks, Region, Main champion or the amount of points/level you have on that champion. With the creation of AtlasBot I had the means to get all of this data and apply it to a user automatically, that's where the idea began.

In the future, AtlasBot is going to evolve into a fully modular bot that will be supplied towards Atlas Partners. This bot will be able to be customized by the server owner to set up special roles using build in commands. Example: `-SetMastery Leblanc 50000 @50k club`, `-SetLogChannel #BotLog`, `-AddAdmin #Role` or `@user`.

How will AtlasBot work?

Like I said before, every single function will be assignable through a very easy to use commands system. The server owner/admin will sign up for partnership on our website, once this is approve they will be send an invite link for the bot. The bot will look for the owner and await a command from him, the verification command. To make sure that AtlasBot doesn't fall into the wrong hands and to get a closed group of users we want to keep the bot Semi-Private. Once someone lets the bot join their server, there will be a simple 1 command verification, the code for the verification will be send to Bort with an invite link to the server and the general information. Servers are stored by ID so simply trying to spoof someone's name will not work.

Everything else will be done by the server owner or admin themselves, AtlasBot will have its own staff that are able to use admin commands on other servers. Server owners, or Bort, will need to give access to the server in question before it works. All of this access can be remove again by one simple command.

From the User side things are very easy. If they sign up on any of the Atlas or Partnered servers they will be signed up on every single one of them. This means if you are a Bard and Thresh main, you can sign up in Bard mains, receive your ranks on Thresh mains with one easy command and then go to Atlas Matchmaking and still have the same roles. The roles that are on the server will of course not be universal but server specific.

What data does Atlas/AtlasBot store?

Right now Atlas only stores your Riot ID, Region and Discord ID. If you name change we still have your data and everything will still be in place. We cannot do anything harmful with this data, we only can use it to receive stats. Making any modifications to the statistics using RiotAPI or any modified version of it is against their rules and will be seen as illegal.

Later on when Atlas expands, we will be working on getting our own Username and Password system working. In the holidays of 2017 me (Bort) and a friend of mine are planning to set up a similar kind of service but then for Overwatch. This project will include a password system.

How will AtlasBot be build?

Right now AtlasBot is build in C# using Visual Studio 2015 and MSSQL (Build in). This testing phase to plan out the idea will keep using this system.

Once we reform to the next iteration of AtlasBot, we will be working within a closed project in GitHub/Gitlab coding in C# using Visual Studio 2015 and using MSSQL in the MS SQL Server Management Tool. This iteration of AtlasBot will have a Database Access Layer with a repository system. Everything will be nicely stored in class libraries and, hopefully, professionally build. We will be both using RiotSharp and Discord.Net for our APIs to reach both Discords and Riots database.5.

Who will the AtlasBot Team be?

The AtlasBot team will consist of programmers, server admins and media designers. We want to have input from as many sources as possible, to make sure we are doing this right. We will be sure to User test every single command, we want to be absolutely sure everyone can use it. Furthermore, we want it to be easy to set up for server owners and even possibly have a web UI interface. Applications will be open to join the Atlas team but everything will be approved and overseen by me. I will lead the project and try to oversee every change to the system.

2. Analysis

Current state of Bots within the League of Legends community.

Right now people are mostly using Nadeko or Orianna Bot for the role-assignment features. These bots work on either adding all the roles manually or adding the roles itself. This system can be improved by a large margin. Once we get the basics done within discord itself we can move to getting an proper Web UI. People will be able to easily assign roles through an very easy dropdown menu and get a visual representation of their settings.

What will AtlasBot's selling feature be?

With AtlasBot we try to focus on a number of things:

- User Support
- Usability
- Modularity

User Support

AtlasBot will have a staff dedicated to working with the community to improve the bot. We want to look at what steps are needed to take towards both the average user and the server owner. This means supporting any feature that they request or changing any parameter that they don't like. This whole project is completely open source for the reason that we want people to have insight within our project.

Usability

We want to give the user the easiest possible way to set up their bot. This will be able to be doable over Discord or a website. We will gather data from common mistakes within parameters and how people fail to understand a command. This data we will use to make a "mistake allowor", making sure that nearly all people can use the bot. Different languages are not going to be supported from the start but are a thing that our system will be build to handle.

Modularity

Like mentioned many times this is the main focus on the project. We are heavily focusing on making this bot their own by any means. Everything about the bot will be customizable, from the messages that it gives you to the role that corresponds to that rank. We want to give people all kinds of options so that they make the server their own. If a feature is needed to run the server it will be custom build and implemented into AtlasBot (University of League's coach system for example).

3. AtlasBot Design

Database

If the roles from Ranks or Regions are not configured manually, they will be automatically searched for by our system. There is an option to do Simple or Complicated ranking, the difference is that Simple only gives you Diamond, Gold, Silver, etc and complicated will also gather the division number with itself.

Region does not have this kind of feature and will either autosearch or search the database for the added Discord Rank. As there is no defined ruleset for how mastery rank roles are done, they will not have a default value. Once I have talked to server owners, I can work on setting a default role if there is a pattern.

UoL:

- Database of choaches with a main role, and +/- 5 main champions
<http://i.imgur.com/y0SVGJP.png>
- Able to filter this database on role or champion
- Update command to update everyones rank on the server automaticly
- CreateRoles option or on server join.