**COMPSCI5018 MSc IT+ Project**

Essay: An investigation into the viability of developing a botting application in Runescape for the MSc IT+ Project

By

Paul Smith

Student ID: 3070332s

Student E-mail: [3070332s@student.gla.ac.uk](mailto:3070332s@student.gla.ac.uk)



Screenshot source: <https://joshuagornall.medium.com/how-i-made-a-runescape-bot-90248acae34>

Contents

[1. Overview 2](#_Toc201332704)

[2. What is Botting? 3](#_Toc201332705)

[3. What is Runescape? 4](#_Toc201332706)

[3.1. Oldschool Runescape (2007scape) 5](#_Toc201332707)

[3.2. Runescape Private Servers (RSPS) 5](#_Toc201332708)

[3.3. Lost City (2004scape) 5](#_Toc201332709)

[4. Runescape Botting 6](#_Toc201332710)

[4.1. Oldschool Runescape Botting 6](#_Toc201332711)

[5. Botting for IT+ Project 7](#_Toc201332712)

[5.1. Project Aims 7](#_Toc201332713)

[5.2. Project Scoping 8](#_Toc201332714)

[5.3. Project Requirements 8](#_Toc201332715)

[5.4. Ethical Concerns 10](#_Toc201332716)

[5.5. Safeguards 11](#_Toc201332717)

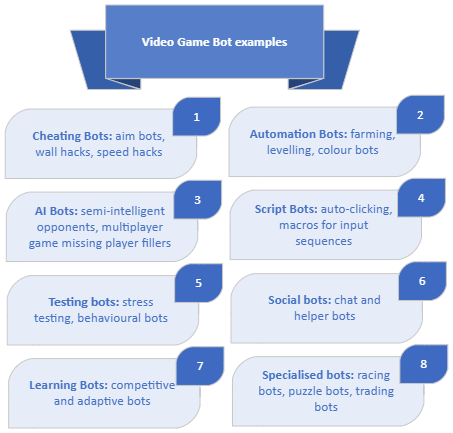
[6. References 12](#_Toc201332718)

[7. Appendices 13](#_Toc201332719)

1. Overview

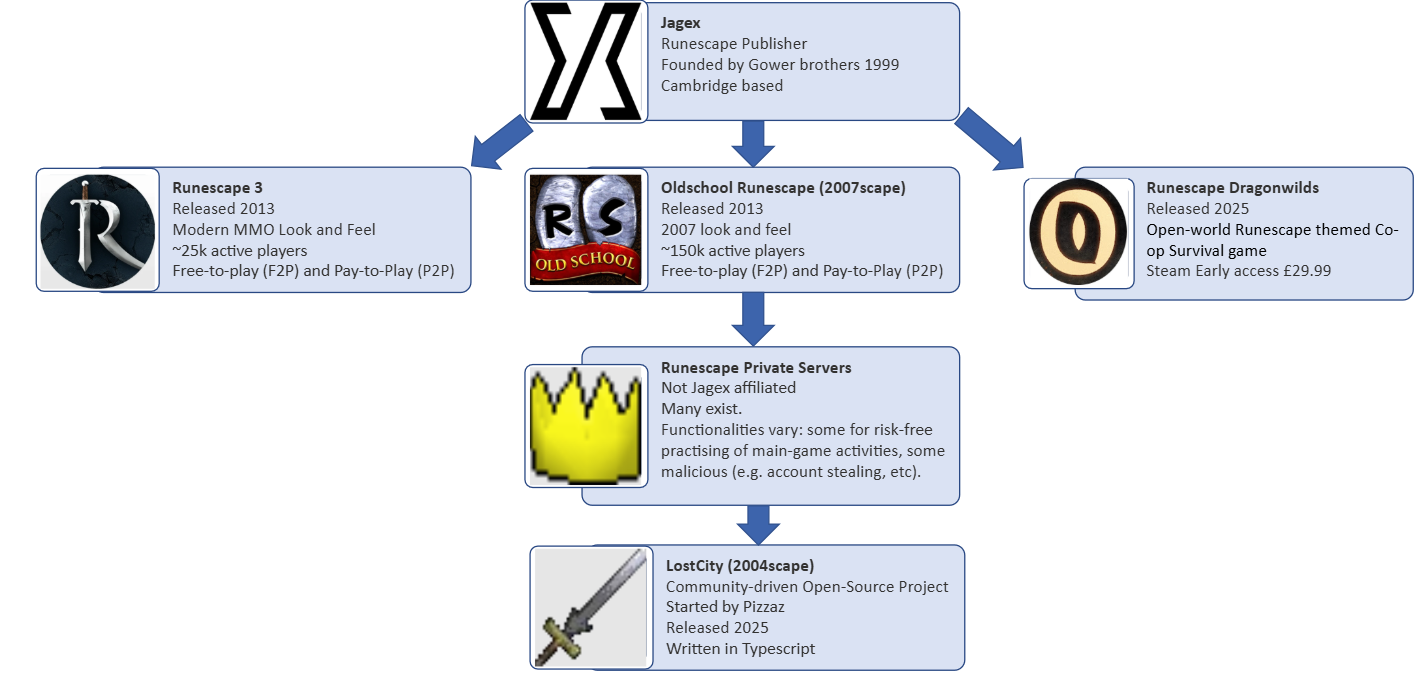
The author must complete an IT+ Project which involves producing a Software artefact. This software artefact will be a bot for Runescape. Sections 1-4 cover contextual information for the game and botting. The structure of the essay is as follows: (2) covers what botting is, why people bot, how people bot; (3) covers what Runescape is, the author’s summary of the game, the different versions of the game; (4) covers Botting within Runescape, botting within Old School Runescape; (5) covers initial project scoping, aims, requirements for Botting, potential ethical concerns and safeguards.

1. What is Botting?

To understand the area of botting, we must first conceptualise botting. Amazon define botting as “an automated software application that performs repetitive tasks over a network… can run independently without human intervention… most bots are useful, outside parties design some bots with malicious intent. Organizations secure their systems from malicious bots and use helpful bots for increased operational efficiency” [AWS]. Others define botting as the ability to “…automatize malicious tasks obtaining some rewards with respect to other game players (the game user increases personal benefits and popularity with low effort)” [Bernardi et al. 2017]. For an overview of the different types of bots, see “Video Game Bot examples figure”. For this project’s botting conceptual definition, we define botting as the ability to automatically perform a sequence of events in-game to the benefit of the botting player or advancement of their goal(s).

1. What is Runescape?

Runescape is a fantasy massively multiplayer online role-playing game (MMORPG). Jagex originally launched Runescape in 2001. Various versions have existed throughout the years (see Runscape versions table). The versions of the game we are concerned with (Oldschool Runescape) is a tile-based and predominantly point-and-click to interact game. The community had an appetite for an earlier version of the game. In March 2013, Jagex released a back-up version of Runescape from 2007 (AKA Oldschool Runescape, Oldschool, OS, OSRS, 2007scape, 07scape, 07). Some unaffiliated community projects have sought to launch their own versions of the game, e.g. LostCity. The focus of this essay will be Oldschool Runescape and LostCity.



|  |  |  |
| --- | --- | --- |
| **Runescape Versions** | | |
| **Version** | **Launched** | **Status** |
| DeviousMUD | 1999 | Defunct |
| Runescape Classic | 2001 | Defunct |
| Runescape 2 | 2004 | Defunct 2013 |
| Oldschool Runescape | 2013 | Live |
| Runescape 3 | 2013 | Live |
| Runescape Private Servers (various) | Various | Various |
| LostCity (a Private Server) | 2025 | Live |

* 1. Runescape Private Servers (RSPS)

RSPS are unauthorised attempts to recreate versions of the game, run by third-parties and first emerged in the early-to-mid 2000s when players sought to customise their experience: an earlier version, faster experience rates, adding custom content, etc. Jagex publicly maintain RSPS infringe upon their intellectual property, taking legal action where appropriate. For instance, against for-profit RSPS. RSPS come with potential security risks, e.g. they may pose as an RSPS but be a front for phishing, account hijacking or another malicious activity.

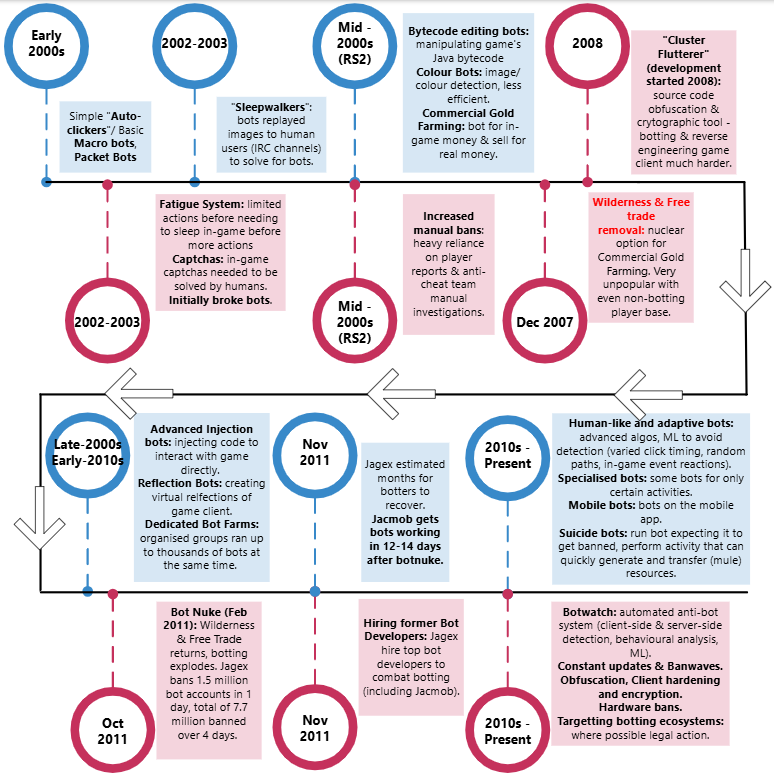
* 1. Lost City (2004scape)

Lost City is a “free, open-source, community-run project” aiming to restore the game from 2004. Players can play on official Lost City servers, or download source code and host their own. As per the developer, Pizzaz, says: “Everything… is completely and transparently open source.” This project will host a local version of Lost City to develop a minimum viable product (MVP) bot.

1. Runescape Botting

The official community wiki defines botting as “using third-party software to perform automated tasks… The terms macroer, autoer, botter or bot usually refer to players who use such programs” []. Since first launching Runescape, Jagex and botters have been in an arms race, with botters automating tasks and Jagex implementing countermeasures (see the Botting timeline for a high-level overview).

Botting timeline:



* 1. Oldschool Runescape Botting

This section will cover the current client types and bot types used. Oldschool Runescape launched when botting on other versions was already well-developed. Since it was an earlier version of the game, some older bots worked in the early days and with a small development team Jagex had few anti-cheat measures, often resorting to manual bans, showcasing this on “bot-busting” livestreams. Today the arms race continues. There are three main client-types: (1) Java, which runs on the Java version of the game, (2) Native, which is the newer client, running on devices natively, (3) Mobile which is the Native client for Android mobile devices only. Known bot types include: (1) Injection, (2) Reflection, (3) Instrumentation, (4) Colour, (5) SRL (Simba Runescape Library), (6) AHK (AutoHotKey), (7) Internal.

|  |  |
| --- | --- |
| Bot Type | Definition |
| Injection | Injects code, allowing client to pull or modify values directly within the game. (Purported fastest and most reliable). |
| Reflection | Similar to injection. Doesn’t directly read values, ‘reflects’ values from specific or known locations. “If injection could only see the client through a mirror”. |
| Instrumentation | Inject a DLL (Dynamic Link Library) while game is running. Frequently used to bot using alternative clients like RuneLite. More complex. Changing game values while it is running. |
| Colour | Uses libraries for colour recognition based on game client visuals to play game. |
| SRL | Allows users to write advanced scripts in a Pascal-like language. Used in OSRS community for decades. |
| AHK | Scripting language to record custom keyboard & mouse shortcuts. Used for macros. Stands for “AutoHotKey”. |
| Internal | For natively run games. Akin to injection |

Source: <https://www.reddit.com/r/RunescapeBotting/wiki/client-list/>

Many reasons exist to use bots within Oldschool Runescape, these include performing repetitive tasks, gold-farming, killing monsters, killing players, levelling player skills and achieving milestones which can take thousands of hours. Some more complex bots are capable of performing multiple activities: killing monsters in an dangerous area where player-versus-player is enabled and switching to killing players when attacked.

Botting use case examples

|  |  |
| --- | --- |
| Use-cases | Benefit(s) |
| Repetitive tasks | Spend less time on dull, boring grinding activities |
| Gold-farming | Obtain resources or gold in-game, sell these for real world currencies. This is known as ‘Real World Trading’ (RWT) and is a bannable offence. |
| Killing Monsters (PVM) | Obtain resources, gold, infamy.  Perform certain activities beyond player’s manual capabilities. |
| Killing Players (PVP) | Obtain resources, gold, infamy.  Perform certain activities beyond player’s manual capabilities. |
| Training skills | Access to content locked behind thousands of hours of grinding. |

Moreover, with incentives for players to bot, there exists an ecosystem of botting platforms, providing players options such as, free-scripts, premium scripts, client-based botting platforms and different types of activities.

Botting provider examples

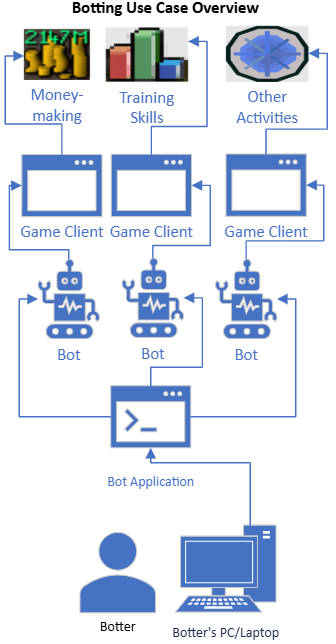
|  |  |
| --- | --- |
| Provider | Type |
| RuneMate | Client-based botting platform. |
| DreamBot | Client-based botting platform. |
| OSBot | Client-based botting platform. |
| TriBot | Client-based botting platform. |
| WASP Scripts | Pixel/Colour-based botting platform. |
| EpicBot | Client-based botting platform. |
| OSMB | Mobile-focused botting platform. |
| Simba/SRL (Simba Runescape Library) | Scripting language and automation tool (mostly pixel based). |
| Synthe.org | Community forums and marketplace. |

This project will investigate the viability of using the colour and SRL approaches. Colour bot examples:



A screenshot of a video game

AI-generated content may be incorrect.

1. Botting for IT+ Project
   1. Project Aims
2. Learn about automation for computer vision through game botting.
3. Produce a Minimum Viable Product (MVP).
4. Determine how to measure success.
5. Demonstrate measurable success.
6. Determine if there’s any benchmarks for bots & APIs.

**Minimum Viable Product (botting application):**

Produce a botting application which contains at least one script. Scripts must use image recognition for input. Scripts must provide output in the form of game-interactions. The botting application must allow the player to bot on at least one account. The botting application must at least work on a locally hosted server of Lost City. The botting application must provide users with a User-Interface to: add a game account to user for botting, monitor status of any currently running bots, show user how long bots have been running for, show user what activities any currently running bots are performing, show user how much progress any currently running bots have made, allow users to stop a given bot or all bots. The botting application must also collect data on the bots for data analysis. Data collected includes: total number of bots currently running, what activities each bot has performed, how long each bot has been running, how much progress each bot has made (e.g. total experience gained, items collected, monsters killed, etc), does the bot appear to be stuck (after a certain period of time elapsed has no further progress been recorded).

**Measuring and demonstrating success:**

This project has the potential for several iterations, each varying in complexity with their own versions of success. Broadly, this project has three main goals.

1. Success through the Minimum Viable Product.
2. Success through botting on Lost City live servers.
3. Success through botting on OSRS.

Goals 2. and 3. are expected to be very difficult due to active sophisticated botting countermeasures

**Measuring success:**

Project success will be predominantly measured through quantitative methods. The botting application MVP will collect data, process it and analyse it to form an internal benchmark. For sufficient data collected, bots will have to have ran for a meaningful time. With only 12-weeks for this project, this emphasises the need for an MVP as soon as possible.

Project Timeline

* 1. Project Scoping
  2. Project Requirements

This section defines some high-level requirements for this project.

**MoSCoW High-level Requirements**

|  |  |  |  |
| --- | --- | --- | --- |
| High-level Requirements | Status | Stage | MoSCoW |
| Bot(s) can run without human intervention. |  | MVP | Must Have |
| Can host game locally. | 18/06/2025 | MVP | Must Have |
| Bot(s) can perform multiple in-game activities. |  | MVP | Must Have |
| Bot application collects data |  | MVP | Must Have |
| Framework for analysing data |  | MVP | Must Have |
| Develop performance metrics |  | MVP | Should Have |
| Image recognition |  | MVP | Must Have |
| Botting Application GUI |  | MVP | Must Have |
| GUI: add account |  | MVP | Must Have |
| GUI: monitor bot(s) status |  | MVP | Must Have |
| GUI: show progress |  | MVP | Must Have |
| GUI: stop bot(s) |  | MVP | Must Have |
| Use MVP to develop API |  | API | Should Have |
| Use API to streamline botscript development |  | API | Should Have |
|  |  | API | Should Have |
| Bot(s) can be ran on Lost City |  | LC | Could Have |
| Botting application can be ran on a Virtual Machine |  | LC | Could Have |
| Bot(s) can be ran on OSRS |  | OSRS | Could Have |
| Botting application can be ran on multiple Virtual Machines |  | OSRS | Would Like to Have |

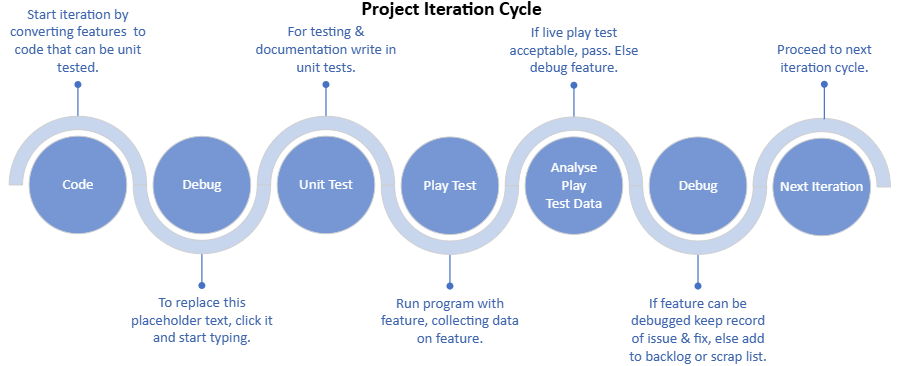
Botting & API Benchmarks:

My proposed benchmarks:

Create my own.

Project Sprint Plan Timeline

Project Overview



Project Design

APIs Explored

|  |  |  |
| --- | --- | --- |
| API | Language | Description |
| RuneLite |  | Java-based |
| DreamBot |  | Java-based |
| OSBot API |  | Java-based |
| TriBot |  | Java-based |
| Altair | Python/JSON | Python-API, outputs JSON description of chart, then rendered by JS frontend, e.g. in HTML. Use when:   * Exploratory Data Analysis (EDA) * Interactive Dashboards * Statistical Visualisation * Need Pandas integration |
| Java AWT | Java | Java’s original API for GUIs. Foundation for Swing. Potential issues:   * Inconsistent Look and Feel * Limited Wedgets * Can feel less responsive. * Custom components can be overly complex |
| SQLite JBDC Driver | Java | Standard Java API defining how java apps connect to and interact with various types of Relational DBs. Some benefits:   * Serverless – can run only on app that uses. * Self-contained: easy to move, copy, backup DB. * Zero-configuration: no installing needed, just link * Small footprint: ideal for embedded systems. |

Libraries Considering

|  |  |  |
| --- | --- | --- |
| Libraries | Language | Description |
| OpenCV | Python | Common infrastructure for Computer Vision (CV) apps. Capabilities include:   * Image and Video processing * Object detection, recognition * Feature extraction, description * Motion estimation and tracking * ML Algos (can work w/ TensorFlow and PyTorche’s DL frameworks * OpenCV(Java) alternative available |
| Tesseract OCR (Pytesseract) | Python | Optical Character Recognition engine. Designed to extract printed or typewritten text from images, docs, etc, for further processing. |
| PyAutoGUI | Python | GUI automation by allowing scripts to control keyboard & mouse. Can obtain screen resolution and colours of specific pixels. |
| Numpy | Python | Can be used for Data Analysis, Science and ML |
| Pillow (PIL Fork) | Python | (Python Imaging Library) Standard image processing tool. Simple API, pixel-level access, cross-platform. Examples:   * Automated Image Processing * Desktop Apps * Data science & ML |
| Sqlite3 | SQL/Python | Good choice, if app needs full-featured SQL DB w/o overhead of separate server process. No configuration needed. Use when:   * Local App Data Storage * Mobile Apps * Prototyping and development * Caching * Lightweight web apps.   Don’t use if:   * High concurrency needed |
| Pandas | Python | Gold-standard for data manipulation, analysis. Examples:   * Data cleaning and preparation * Exploratory Data Analysis * Statistical Analysis * Data Transformation for ML * Data collection |
| Matplotlib/ Seaborn | Python | Matplotlib: vast array of static plots, can be verbose for simple plots.  Seaborn: built on Matplotlib & pandas. Specific focus on statistical data visualisation, can look better than matplotlib, can simplify many complex plots, potentially less lines of code, but less granular than Matplotlib |
| SRL (SIMBA Runescape library) | SIMBA |  |

Frameworks Considering

|  |  |  |
| --- | --- | --- |
| Frameworks | Language | Description |
| RuneLite Plugin Development Framework | Java (mainly) | Open-source, community-driven ecosystem for the OSRS RuneLite client. Plugin Dev Framework is a set of APIs, interfaces, utilities for third-party Devs to create custom plugins. Key benefits:   * Gamestate access, e.g. Player position, stats, inventory * Overlay system: highlight objects, draw custom text, graphics, shapes paths on game screen. * Input simulation * Resource management: manage in-game images, sounds * Data models: in-game entities can be abstracted to work with game data * Open-source * Not intended for bot development |
| DreamBot Framework | Java | Community-driven botting client & dev framework for OSRS. Provides:   * API to interact with most game elements. * Gamestate real-time access * Simulation of human-like input * Efficient task execution * Computer Vision capabilities * Optimised pathfinding system. * Anti-ban compliance features |
| OSBot Framework | Java | * Extensive API for access to gamestate internals * Event-handling * Efficient pathfinding * Not intended for Computer Vision |
| Tribot Scripting Framework | Java | * Gamestate access: primarily injection * Event-driven * Pathfinding * Renders overlays * Anti-ban ability: has strong anti-detection focus |
| Custom Bot Framework | Java and/or Python | This would be if I built my own. Very likely to be far too ambitious for this project. |
| TensorFlow | Python (mainly) | Open-source ML framework. Allows creating, training, deployment of ML models across platforms. Has variety of APIs. Applications include:   * Computer Vision * NLP * Time series forecasting * Robotics and autonomous systems. |
| PyTorch | Python | Leading Open-source ML framework. Example apps:   * Deep Learning Research * Computer Vision * NLP * Reinforcement Learning * Time series forecasting |

Toolkits Explored

|  |  |  |
| --- | --- | --- |
| Toolkit | Language | Description |
| WASP Scripts | Simba Script (mostly) | Simba – Pascal-like language for automation. Mainly used for automation and often with colour-botting methods.  Relies on: computer vision and pixel analysis.   * RuneLite, Dreambot, OSBot are reflection/injection clients that read internal gamestate * WASP Scripts instead read pixels on the screen.   Primary functionality:   * Image recognition, colour matching, mouse & keyboard simulation, can infer gamestate.   Potential issues:   * Gamestate data – no direct access * Fragile – changes in graphics, visual distractions could break bot * Speed – scanning pixels, image recognition could be slower and computationally expensive than direct memory access * Intelligence – more difficult to build from colour alone |
| SIMBA | Pascal | Originated from SCAR (Scripting for Computer Automated Recognition).   * Core program written in Pascal. * Open-source automation tool. * Tool to control through simulating mouse & keyboard input with advanced image and colour recognition. * Treats the game client as a black box. * Client flexibility – in theory work with any client if visuals same. * “Hardcore” anti-detection – historically claimed harder to detect because purely visual. * Education for Computer Vision – good way to learn CV for automation. * Fragile to visual changes * Performance intensive * Dynamic environments – areas where visuals are obscured expect lower reliability |
| DreamBot | Java | Commercial.   * Deep game injection and reflection – real-time access to gamestate data. * Facilitates highly humanised input simulation * Versatility – deep control allows more wide range of bots. * Extensive API for Game interactions * Can develop custom graphical overlays * Advanced pathfinding |
| OSMB | Python | Niche focus on OSRS mobile client.   * I would struggle to test on Lost City * Framework sends commands to Android emulator or connected Android device to simulate input. * Primarily based on computer vision * Provides a Python Scripting API * Alleges that it can be ran 24/7 without bans |
| Swing | Java | Java GUI toolkit. Features:   * Wide component variety * Object-oriented & Builds on AWT * MVC Architecture * Layout Manger * Lightweight look & feel |

* 1. Ethical Concerns

A number of ethical concerns around botting can be raised. Broadly these concerns would fall into categories of for and against botting. Examples arguments:

1. Using bots shallows achievements:
   1. Botting players who would otherwise put in 1000 hours of manually grinding away at a task to achieve the satisfaction at the end will no longer
   2. Genuine players have their genuine achievements ‘shallowed’ or ‘inflated’ through others using bots, disincentivising genuine players to pursue goals.
2. Playing with bots is bad for the game experience:
   1. Playing with bots won’t have the same level of interaction as would playing with genuine players.
   2. Playing with bots ‘crowds-out’ in-game non-instanced activities.
   3. Playing with bots become stale.
3. Bots are bad for the game state:
   1. Bots (especially bot farms), have the potential to flood the game with resources, inflating the economy.
   2. Bots potentially damage the intellectual property of the company (game economy).
4. Bots are an economical solution:
   1. Players play this game with their free-time, of which they only have so much. Bots allow them to make more use of their time.
   2. Bots potentially allow players to allocate time away from tedious, dull in-game tasks, instead focusing towards they are interested in (maybe ones bots haven’t yet been developed for).
   3. Safeguards

The author has obtained and will proceed with the explicit understanding of his supervisor that the development of any botting scripts is not intended for malicious purposes.

The author understands that the development of bots is to proceed with the goal of having learned something through this 12-week process.

1. References

Botting Definitions

<https://aws.amazon.com/what-is/bot/>

Mario Luca Bernardi, Marta Cimitile, Fabio Martinelli, and Francesco Mercaldo. 2017. A time series classification approach to game bot detection. In Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics (WIMS '17). Association for Computing Machinery, New York, NY, USA, Article 6, 1–11. <https://doi.org/10.1145/3102254.3102263>

<https://oldschool.runescape.wiki/w/Botting>

Official Runescape sources:

[1] <https://play.runescape.com/>

[2] <https://oldschool.runescape.com/launcher#_ga=2.26562621.1410479580.1750164561-1696873367.1750164559>

<https://oldschool.runescape.com/>

<https://www.jagex.com/>

Runescape Community sources:

<https://oldschool.runescape.wiki/>

<https://oldschool.runescape.wiki/w/Project_Zanaris>

<https://oldschool.runescape.wiki/w/Code_of_Conduct>

<https://legal.jagex.com/docs/terms/terms-and-conditions>

<https://2004.lostcity.rs/>

<https://github.com/LostCityRS/Server>

Runescape Community population sources:

<https://www.misplaceditems.com/rs_tools/graph/>

<https://mmo-population.com/game/runescape>

Runescape Gold sites

Runescape Botting sites

Runescape & Venezuelans

<https://www.npr.org/transcripts/1018915121>

<https://www.investmentmonitor.ai/tech/gaming-venezuela-runescape-wages-digital/>

Jagex and the Botting arms Race:

<https://www.pcgamer.com/runescape-bot-nuking-event-bans-1-5-million-bots-in-one-day/>

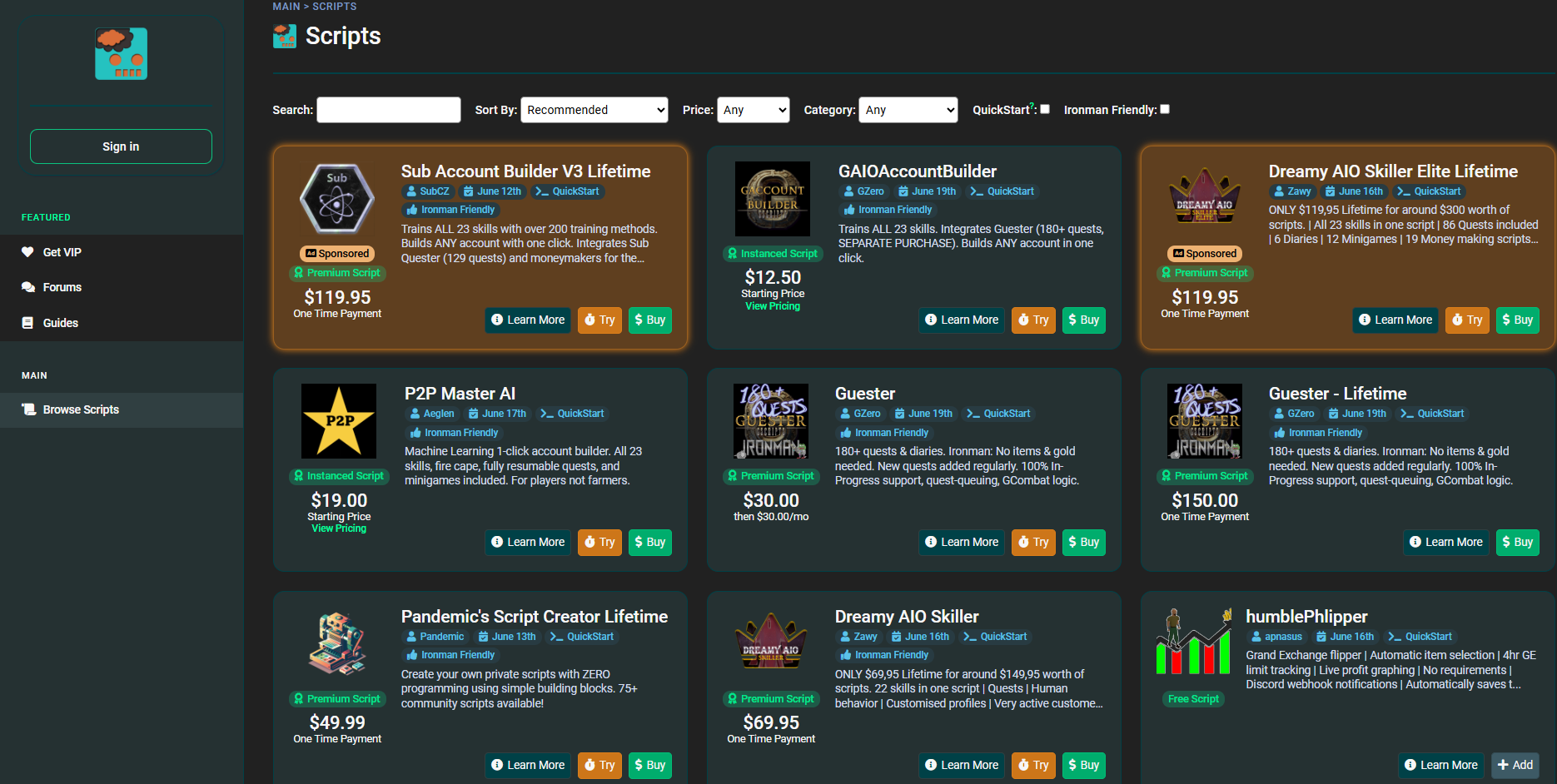
<https://www.youtube.com/watch?v=VZHalHeWer8>

Botting example:

<https://joshuagornall.medium.com/how-i-made-a-runescape-bot-90248acae34>

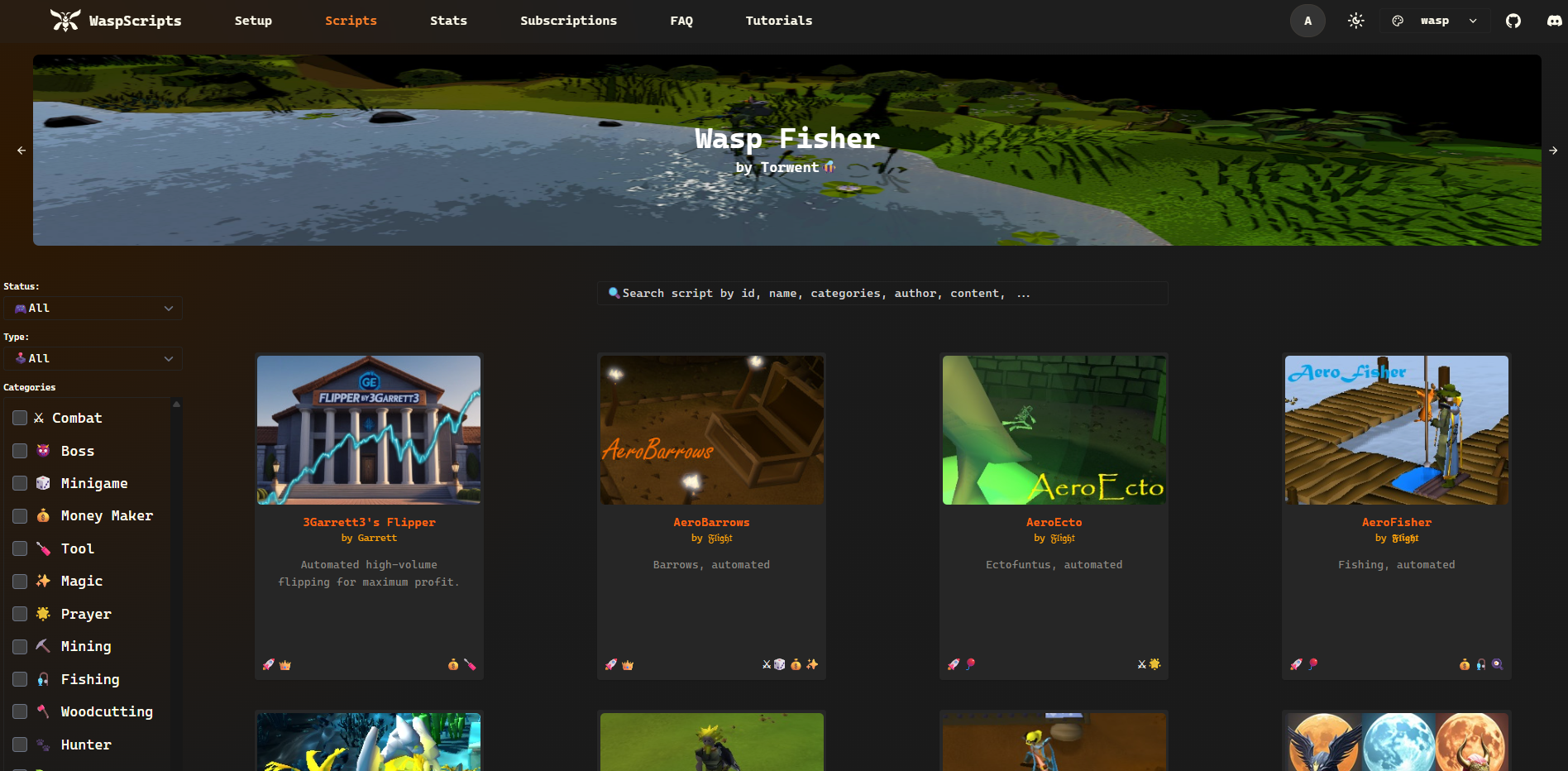
1. Appendices

Appendix A – Dreambot Scripts online store screenshot:



Obtained 20/06/2025 from: <https://sdn.dreambot.org/scripts>

Appendix B – Wasp Scripts online store screenshot:



Obtained 20/06/2025 from: <https://waspscripts.com/scripts>

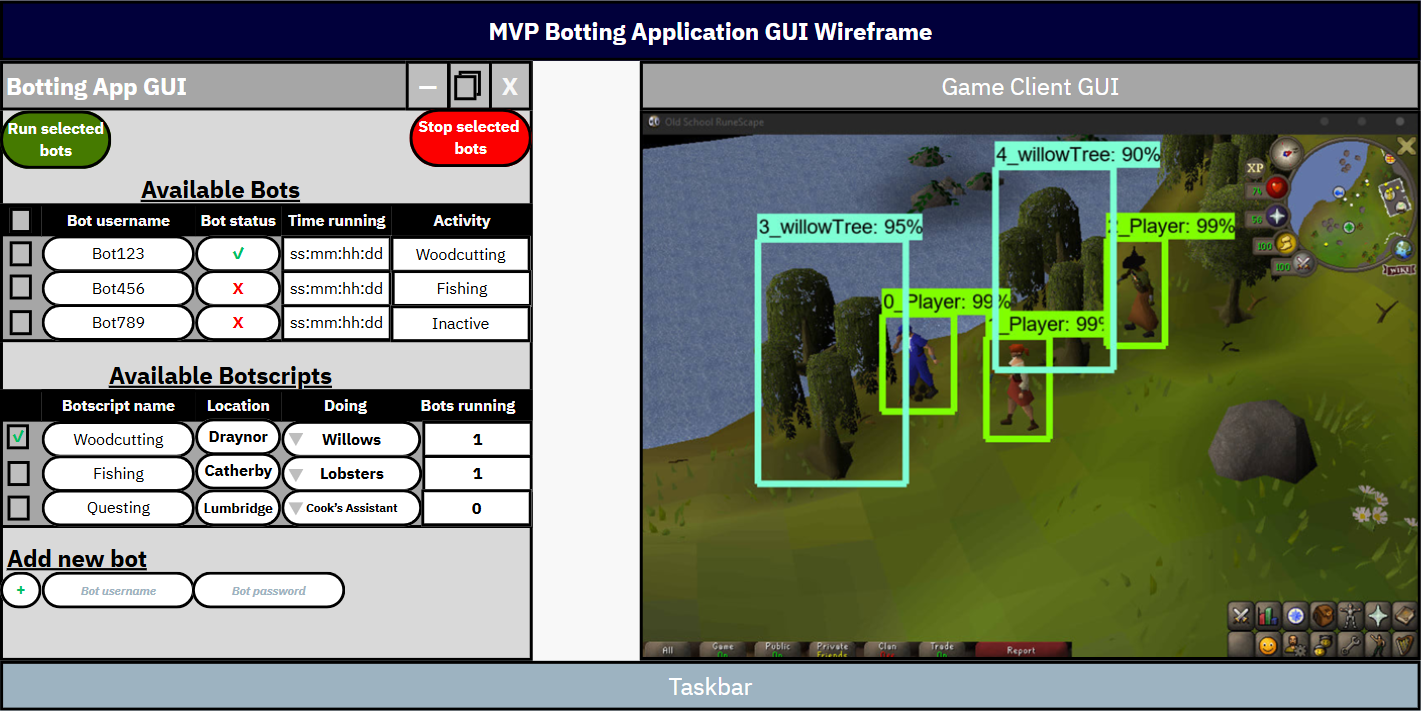
Appendix C – Runemate Scripts online store screenshot:

A screenshot of a computer

AI-generated content may be incorrect.

Obtained 20/06/2025 from: <https://www.runemate.com/store>

Appendix D – Wireframe for MVP Botting Application GUI





Appendix F – High-level project sequence diagram

A diagram of a software development

AI-generated content may be incorrect.