Playing the Turing Game (NAG-21-266)

November 2021

By

Callum Gray

Student number 201915545

Word count: 3289

Contents

1. Project background and purpose 3
1.1. Objectives 3
1.2. Scope 4
1.3. Deliverables 4
1.4. Constraints 4
1.5. Assumptions 5
2. Project rationale and operation 6
2.1. Project benefits 6
2.2. Project operation 6
2.3. Options 6
2.4. Risk analysis 7
2.5. Resources required 8
3. Project methodology and outcomes 9
3.1. Initial project plan 9
3.1.1. Tasks and milestones 9
3.1.2. Schedule Gantt chart 11
3.2. Project control 12
3.3. Project evaluation 13
4. Appendix A 14
5. Appendix B 15
6. Bibliography 16

1. Project background and purpose

The Turing game is a test created and named after Allan Turing. He created this test as a means of testing whether an artificial intelligence (AI) could be considered conscious, or to be able to exhibit sufficient intelligence in order to compete with a human being. The Game is made up of a conversation between two or more entities, one of the entities must be a human and the other entities can be either a human or an AI. The conversation consists of a series of questions posed by the human that the other entity must answer. The AI is deemed to have passed the test if the human is unable to reliably identify it as not human based on its responses.

The aim of this project is to create a framework of tools that can be used to allow users to log into a system and play the Turing game against other users or an AI. This should be designed with considerations to make the game portable so it could be used as a teaching aid in schools.

The framework of tools to be created will allow multiple users to connect to some form of server that shall host the chat bot and allocate the connected players to a room with either a chat bot or another user.

1.1. Objectives

This project is useful as it creates a tool for use in educational visits or as a general teaching aid to teach children about AI and how it works in an interactive way.

The project will have primary objectives and some secondary objectives. The primary objectives will be completed in full and will be a measure of the project's success. The secondary objectives will be stretch targets achieved only if time allows. The primary objectives are as follows:

1. Design and create a UI for the tool that the end user will interact with - this will give the user the functionality to send and receive messages as well as buttons to enter their predictions of what they're talking to
2. Produce a chat bot - create a chat bot that will be hosted on the server and give responses
3. Produce a server-side tool - this will allocate the users that connect to it into separate rooms
4. Produce a Complete set of Documentation to accompany the software such as Requirement specifications and Design descriptions.
5. Carry out user evaluation to test the solution – the users shall fill out two feedback forms see appendices for details.

The secondary objectives are as follows:

1. Create a mobile app version of the User tool – this will allow the system to become more portable
2. Incorporate machine learning into the chat bot – this will allow it to learn better responses from the users who log in

1.2. Scope

The scope of this project includes the design and development of a framework of tools that can be used to play the Turing game. The framework created as part of this project shall be designed to allow the frameworks to be implemented later.

The scope of this project does not include the procurement and provision of the hardware to host the software and run the game. As this will put un-necessary constraints on the project as well as limit its potential use and benefit.

1.3. Deliverables

The artefacts that will be created to be delivered upon completion of the project will consist of 3 individual components:

1. A Server-side tool that shall provide the capability to manage the “rooms” that people can enter to play the game and which room hosts the chat bots
2. A Client-side tool that shall provide the capability for users to send and receive messages as well as input their choices.
3. A Chatbot tool that shall provide the capability to read and respond to the messages that it is sent as well as ask questions
4. Project report including requirements, design, code, testing and results
5. Project plan detailing the true results vs the estimates
6. Risk analysis to allow for risk mitigation and minimise project impacts
7. A demonstration capturing the functionality of the solution created.

These are the bare minimum deliverables required to consider the project complete. Each component will be assessed against their requirements to establish whether it has been completed successfully.

1.4. Constraints

The project must comply with the both the Hull University data protection policy and the Hull University research ethics policy.

The Hull University data protection policy applies to all staff, students, contractors, and volunteers working for the University and ensures that they comply with the Data Protection Act 2018. This policy may impact upon the research required for the project. If any breach of this policy is committed it may need be considered under the Student Disciplinary Regulations.

The Hull University research ethics policy is in place to ensures that the academic conduct of all research conforms to the highest possible standards. This can result in additional time needed for the project as an ethics check list will have to be carried out before starting any research.

Time constraints are a consideration as this project will need to be completed in a strict time frame alongside other coursework and work commitments that will need to be managed. One of these other commitments is that during university holidays there will be little time available to work on the project. This is because the Project manager is employed by BAE systems who requires them to work to work full time during holiday periods.

Another consideration is that time is needed to complete two other units in the first trimester and a second set of 2 units in the second trimester of the course. Managing time effectively will be critical for the success of this project.

The chatbot will need to respond in real time with an appropriate level of delay so as not to be obviously automated as this would defeat the purpose of the game this will require thorough research and testing to achieve.

Additional constraints include the ongoing global pandemic which has resulted in two national lockdowns. If another national lock down should occur it could have an adverse impact upon the testing of the project as getting the solution to the testers may become difficult.

1.5. Assumptions

At this stage of the project certain assumptions need to be made to proceed, these are:

- The equipment that is going to be used will be in working condition throughout the project life cycle. However, there could be issues where the resources become unavailable due to malfunctions or damage during the project.
- End users will be available in the time allocated for testing as part of the project plan. As each End user tester will likely have their own project or work to work on their may be conflicts in terms of available time.
- The API being used will have sufficient capability to complete the project and will provide adequate functionality.
- The predicted project's timeline can be met, and the project will be completed within the allocated time.
- It will be assumed that 20 hours will be available per week to work on the project

2. Project rationale and operation

2.1. Project benefits

This project will be useful in creating a tool that can be used as a teaching aid in schools or on educational visits. Teachers can use it to aid in teaching students about AI and the principles behind the Turing game. The project also offers a learning Opportunity for the project author in researching and better understanding AI principles and demonstrating the capabilities of modern computing.

2.2. Project operation

The Project shall be managed using an AGILE management plan. In order to achieve this, it will be broken down into sprints consisting of 3 weeks each sprint will consist of task allocation at the beginning task completion during the remainder of the sprint and a retrospective review of the workflow for the sprint at the end. This will allow for the project to be dynamically restructured following the reviews in ways to improve the workflow and efficiency of the project.

To manage the content of the project, a form of Source control will be implemented.

2.3. Options

There are many options available for the project these have been broken down into several different types:

Languages: **C#**, C++, java, JavaScript, **python**

Development Techniques: Inheritance, **Object Oriented Programming(OOP),** Entity Component System(ECS)

Source Control: CloudForge, Team Foundation Server(TFS), Bitbucket, GitLab, **Git Hub**

Project Planning: TFS , Jira, GitHub, **Microsoft Project**, Trello

Design: Microsoft Visio, **Lucid Chart**, PTC Modeler

User Interface Design: **MockFlow**, adobe illustrator, Ms paint

Integrated Development environment (IDE): **Microsoft visual studio 2019**, JetBrains, PyCharm, **notepad++**

There are a lot of options available for use in the project. All the mentioned options provide the necessary capabilities for completing the project, however due to the strict time constraints it is necessary to base the decisions mainly on the familiarity the project manager has with the relevant software. This is because learning how to use a new piece of technology or write in a new language would not be feasible with the timeline and research necessary for the project to complete. The options that will be used have been highlighted. Where multiple options are highlighted, it may be due to the different uses for either option. For example, C# will be used as the main language for the solution however python can be used to create very powerful test scripts that are light weight.

2.4. Risk analysis

The following table is used to rate the risks to the project's success

Risk	Mitigation	Likelihood	Severity	Impact
Computer Failure	All work I backed up in two locations a physical external SSD and remotely to a GitHub repository.	Low	medium	The impact on the project should be low. Due to the nature of the project, it can be transferred from a backup onto a university pc to be completed.
Illness for more than 1 week	The project will be able to be accessed from home to work on it if the illness is not too severe	low	medium	The impact of this risk could be severe depending upon the severity of the illness as long as it is not too severe work can be carried out from home
Lock down	All work is set up to allow remote/homeworking including contact with project supervisors	medium	low	The impact on the project will be minimal due to the nature of the changes to working practices put in place as a result of previous lockdowns.
Hard drive failure	All work I backed up in two locations a physical external SSD and remotely to a GitHub repository.	Low	low	The impact on the project should be low. Due to the nature of the project, it can be transferred from a backup onto a university pc to be completed.
File corruption	All work I backed up in two locations a physical external SSD and remotely to a GitHub repository.	Low	low	The impact on the project should be low. Due to the nature of the project, it can be transferred from a backup onto a university pc to be completed.
Failing to meet time constraints	Using an agile project management system with 2 week sprints the project can be dynamically re-scoped to better suit workflow in order to avoid things dragging beyond their time frame	medium	medium	The impact of not meeting timelines is fairly severe as depending upon which task is falling behind their may be little to no room for recovery leading to a knock-on effect.

2.5. Resources required

The project will use standard resources that are readily available and can be run on a windows 10 computer with minimal set up required. To make and edit the source code for the project a several IDE (Integrated development environment) shall be used these include:

- notepad++
- Visual studio 2019

These programs will be used to create the code for the server-side tool, the client-side tool, the chat bot and to produce The UI for the client-side tool as well as a basic interface for the server side to allow ease of editing.

Microsoft Office suite will be needed for 2 of the products included these are as follows:

- Microsoft Word shall be used In order to produce the documentation such as reports, reviews, SRS(software requirements specifications), SDD(Software design description), STD(Software test description). These documents will be submitted alongside the artefact that is created for the final submission to help justify the completeness of the project.
- Microsoft Project will be used to manage the Project and track the work that needs to be done.

Git hub shall be used to store and manage the source code and documentation. This will allow for change management and back us as well as being able to restore any file corruption or errors to a previous state.

An external SSD shall also be used as a back up as this will allow the documentation and source code to be restored regardless of internet connection.

3. Project methodology and outcomes

3.1. Initial project plan

3.1.1. Tasks and milestones

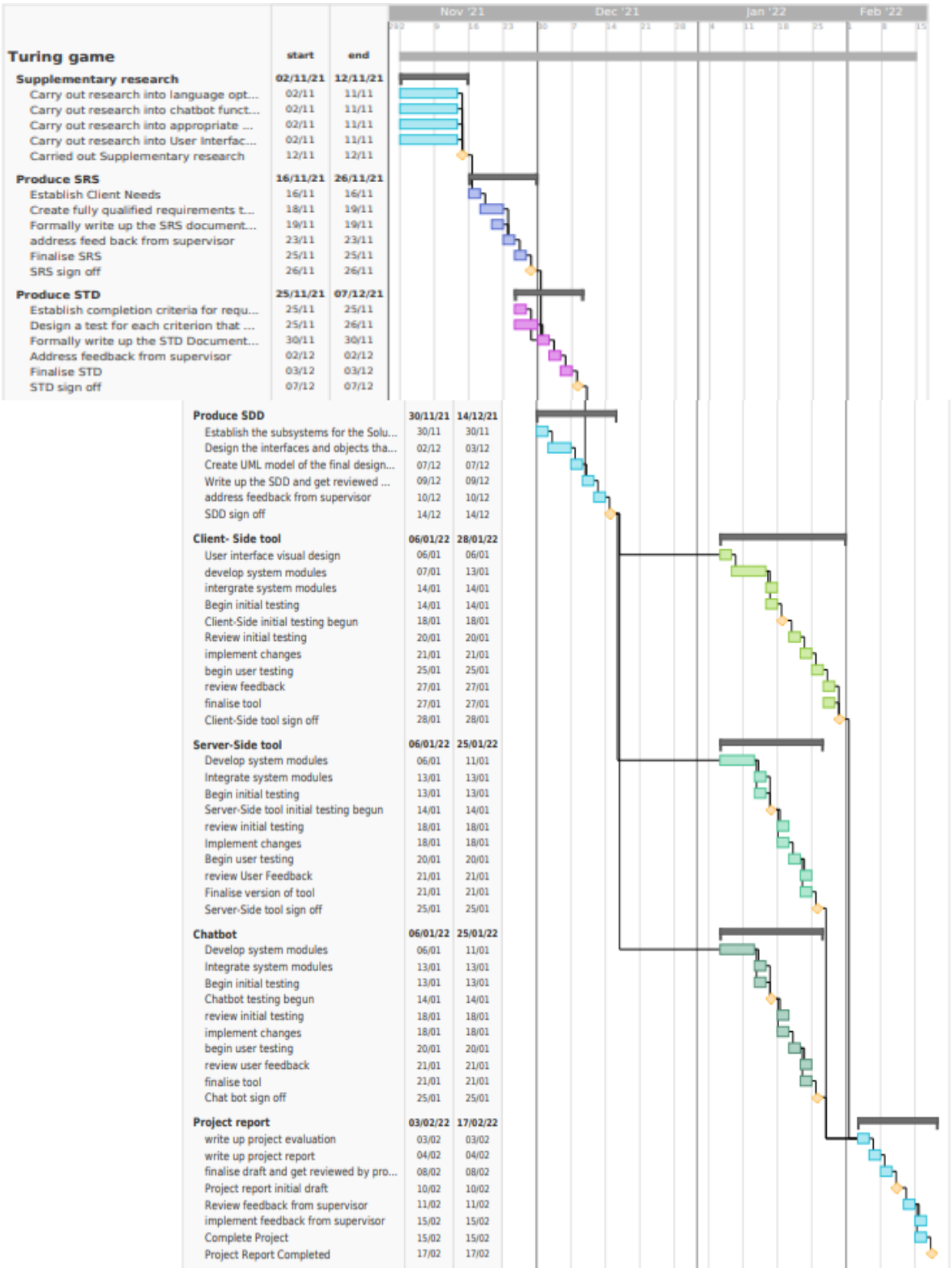
The following denotes the milestones and what tasks are required to meet those milestones duration is in hours with an assumed availability of 24 hours available per week

Milestones:

Mile stones:	Tasks:		Hours:
1. Carry out Supplementary research	Total:		40
	1.1.	Carry out research into language options	12
	1.2.	Carry out research into chatbot functionality and setup	12
	1.3.	Carry out research into appropriate delay times for responses	8
	1.4.	Carry out research into User Interface library	8
2. SRS – signed off	Total:		40
	2.1.	Establish client needs	6
	2.2.	create fully qualified requirements that meet client needs	14
	2.3.	Formally write up the SRS document and get reviewed by Project supervisor	8
	2.4.	address any feedback from supervisor	8
	2.5.	Finalise SRS	4
3. STD– signed off	Total:		40
	3.1.	Establish completion criteria for requirements	4
	3.2.	Design a test for each criterion that satisfies a requirement	16
	3.3.	Formally write up the STD Document and get reviewed by Project Supervisor	8
	3.4.	Address feedback from supervisor	8
	3.5.	Finalise STD	4
4. SDD– signed off			60
	4.1.	Establish the subsystems for the Solution	8
	4.2.	Design the interfaces and objects that will need to be created	20
	4.3.	Create UML model of the Final designs for each subsystem	8
	4.4.	Write up The SDD adding in descriptions for each entity in the UML diagram such as classes functions variables interfaces etc. and get reviewed by project supervisor	12
	4.5.	Address supervisor feedback	8
	4.6.	Finalise STD	4
5. Client-Side tool – initial testing begun	Total:		36
	5.1.	User interface visual design	6
	5.2.	Develop system modules	24
	5.3.	Integrate system modules	4
	5.4.	Begin initial testing	2
6. Server-Side tool – initial testing begun	Total:		30
	6.1.	Develop system modules	24
	6.2.	Integrate system modules	4
	6.3.	Begin initial testing	2
7. Chatbot – initial testing begun	Total:		30
	7.1.	Develop system modules	24
	7.2.	Integrate system modules	4

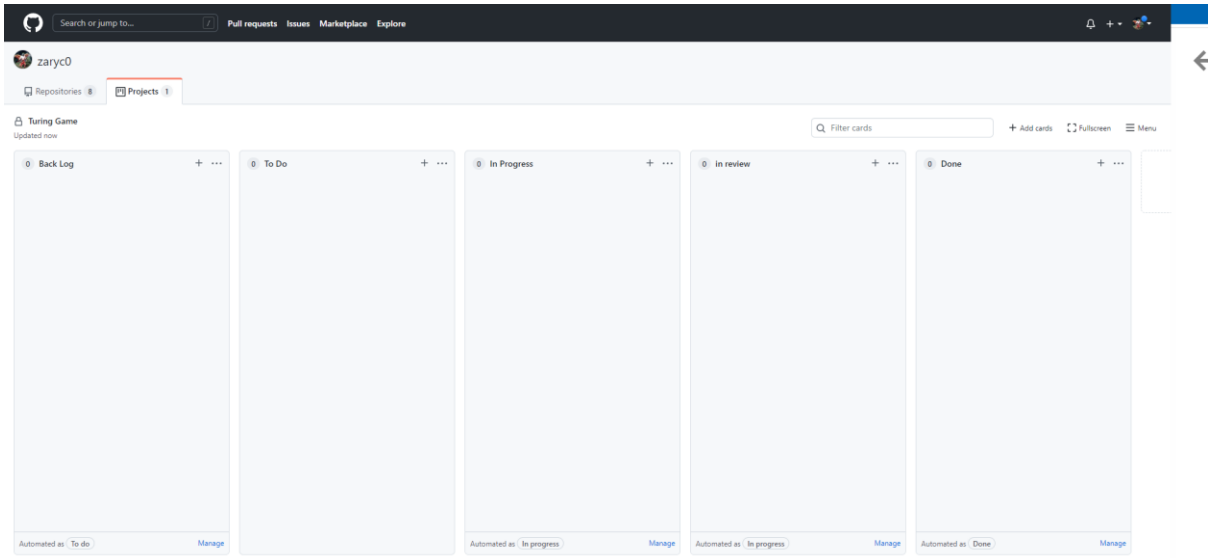
Mile stones:	Tasks:		Hours:
	7.3.	Begin initial testing	2
8. Client-Side tool – Sign off	Total:		30
	8.1.	Review initial testing	4
	8.2.	Implement changes	6
	8.3.	Begin User testing	8
	8.4.	Review user feedback	4
	8.5.	Finalise version of tool	4
9. Server-Side tool – Sign off	Total		30
	9.1.	Review initial testing	4
	9.2.	Implement changes	6
	9.3.	Begin User testing	8
	9.4.	Review user feedback	4
	9.5.	Finalise version of tool	4
10. Chatbot – Sign off	Total:		30
	10.1.	Review initial testing	4
	10.2.	Implement changes	6
	10.3.	Begin User testing	8
	10.4.	Review user feedback	4
	10.5.	Finalise version of tool	4
11. project report– initial draft	Total:		48
	11.1.	write up project evaluation	20
	11.2.	write up project report	20
	11.3.	finalise draft and get reviewed by project supervisor	8
12. Project report – sign off	Total:		20
	12.1.	Review feedback from project supervisor	4
	12.2.	Implement feedback from supervisor	8
	12.3.	Complete report	8
TOTAL	Total:		

3.1.2. Schedule Gantt chart



Turing game	start	end
Supplementary research	**02/11/21**	**12/11/21**
Carry out research into language opt...	02/11	11/11
Carry out research into chatbot funct...	02/11	11/11
Carry out research into appropriate ...	02/11	11/11
Carry out research into User Interfac...	02/11	11/11
Carried out Supplementary research	12/11	12/11
Produce SRS	**16/11/21**	**26/11/21**
Establish Client Needs	16/11	16/11
Create fully qualified requirements t...	18/11	19/11
Formally write up the SRS document...	19/11	19/11
address feed back from supervisor	23/11	23/11
Finalise SRS	25/11	25/11
SRS sign off	26/11	26/11
Produce STD	**25/11/21**	**07/12/21**
Establish completion criteria for requ...	25/11	25/11
Design a test for each criterion that ...	25/11	26/11
Formally write up the STD Document...	30/11	30/11
Address feedback from supervisor	02/12	02/12
Finalise STD	03/12	03/12
STD sign off	07/12	07/12
Produce SDD	**30/11/21**	**14/12/21**
Establish the subsystems for the Solu...	30/11	30/11
Design the interfaces and objects tha...	02/12	03/12
Create UML model of the final design...	07/12	07/12
Write up the SDD and get reviewed ...	09/12	09/12
address feedback from supervisor	10/12	10/12
SDD sign off	14/12	14/12
Client- Side tool	**06/01/22**	**28/01/22**
User interface visual design	06/01	06/01
develop system modules	07/01	13/01
intergrate system modules	14/01	14/01
Begin initial testing	14/01	14/01
Client-Side initial testing begun	18/01	18/01
Review initial testing	20/01	20/01
implement changes	21/01	21/01
begin user testing	25/01	25/01
review feedback	27/01	27/01
finalise tool	27/01	27/01
Client-Side tool sign off	28/01	28/01
Server-Side tool	**06/01/22**	**25/01/22**
Develop system modules	06/01	11/01
Integrate system modules	13/01	13/01
Begin initial testing	13/01	13/01
Server-Side tool initial testing begun	14/01	14/01
review initial testing	18/01	18/01
Implement changes	18/01	18/01
Begin user testing	20/01	20/01
review User Feedback	21/01	21/01
Finalise version of tool	21/01	21/01
Server-Side tool sign off	25/01	25/01
Chatbot	**06/01/22**	**25/01/22**
Develop system modules	06/01	11/01
Integrate system modules	13/01	13/01
Begin initial testing	13/01	13/01
Chatbot testing begun	14/01	14/01
review initial testing	18/01	18/01
implement changes	18/01	18/01
begin user testing	20/01	20/01
review user feedback	21/01	21/01
finalise tool	21/01	21/01
Chat bot sign off	25/01	25/01
Project report	**03/02/22**	**17/02/22**
write up project evaluation	03/02	03/02
write up project report	04/02	04/02
finalise draft and get reviewed by pro...	08/02	08/02
Project report initial draft	10/02	10/02
Review feedback from supervisor	11/02	11/02
implement feedback from supervisor	15/02	15/02
Complete Project	15/02	15/02
Project Report Completed	17/02	17/02

3.2. Project control



The project will be run using sprints the sprints will consist of 2 weeks of work followed by a review of the work completed in the current sprint this will allow the project to be constantly reviewed and improved over the project lifecycle. The weekly meetings with the project supervisor will also be a valuable opportunity to review the progress made on the project. The success of this control method can be reviewed by analysing how closely the workflow matches the predicted plan. As can be seen in the image below the work will be tracked in 5 stages, these are:

- The Backlog Column – this is where all the project tasks will start their life at the beginning of the project.
- The To Do Column – This is where the tasks that are needed to be completed in the current sprint will be moved to.
- The In Progress Column – This is where the tasks that are currently being worked on will be moved to.
- The In Review Column – This is the column where work will be moved to when it needs to be reviewed pull requests will also automatically populate into this column.
- The Done Column – This column is where the completed tasks will be moved to these will be reviewed at the end of the each sprint to assess the work flow.

3.3. Project evaluation

The projects Artefact will be evaluated by assessing it against the requirements established at the beginning of this project. This will give a definite evaluation of the success of the artefact. The Documentation will be evaluated by a review process that will be ongoing. By using the Agile method and sprints the project will undergo an ongoing review every two weeks to evaluate the success up to the point of each review. This will give Opportunity to propose adjustments that will need to be made and any improvements that could improve workflow.

As part of the testing There will be User evaluation the feedback from this will be in the form of 3 review Sheets (see appendix A and B) and the results of these will be considered for improving the Artefact that is created for the Project.

4. Appendix A

User review Form 1 - UI

User Feedback Form UI review		
Criterion:	**Feedback:**	**Score (1 -10)**
How would you rate the responsiveness of the UI?		
How would you rate the appearance of the UI?		
How easy was the Application to use? Is it intuitive or could it be improved?		
Any other Feed back?		

5. Appendix B

User review Form 2 – game review

User Feedback Form game review		
Criterion:	**Feedback:**	**Score (1 -10)**
How would you rate the responsiveness of the application ? e.g are the replies quick? Does it take long to connect to another user etc.		
How obvious is the chat bot? e.g. is it replying too quickly? Is it replying too slowly? Does it ask appropriate questions		
Any other Feed back?		

6. Bibliography

Copeland, B. J. (2005). The Essential Turing: The ideas that gave birth to the computer age. *Oxford: Oxford University Press*.

Turing, A. (1950). *Computing Machinery and Intelligence.* Manchester.