[Slide 1 - Title]

Background

[Slide 2 - Introduction]

\* Computerised devices and the challenge of file organisation

\* Benefits of intelligent agents in processing and analysing data

\* Vulnerability to malicious files

[Slide 3 - Objectives]

Objectives:

1. Design and develop intelligent agents for file classification

2. Develop an interface for user interaction and feedback

3. Perform testing and validation for accuracy and efficiency

[Slide 4 - Methodology]

\* Multi-agent system for file classification

\* Collaboration and interaction among agents

\* Use of reactive agents for fast and efficient response

[Slide 5 - Project Design]

\* Use case diagram UML for agent roles and interactions

\* Sequence diagram UML for object interactions and order

[Slide 6 - Development]

\* UploadAgent: Secure file uploads

\* ClassifierAgent: Classify files based on extensions

\* MoveFileAgent: Move files to designated folders

\* ViewFileAgent: Display files to the user

[Slide 7 - Testing and Evaluation]

\* User interface for file upload

\* User interface displaying file explorer

\* Interface displaying classification results

[Slide 8 - Programming Approach]

\* Programming approach: Python programming

\* Flask: Lightweight framework for building web interfaces

\* Werkzeug: Ensures security in file handling and communication

\* os: Handles operating system-dependent functionalities

\* shutil: Performs high-level file operations and management

[Slide 9 - Challenges]

\* Accurate file classification

\* Secure file upload to prevent attacks

[Slide 10 - Conclusion]

\* Proposed plan for designing intelligent agents for file classification

\* Benefits of automation in file organisation and user experienc