RBS, WBS and schedule

Team 14

目录

RBS (requirement breakdown structure) 2

System Functional Requirements 2

Performance Requirements 3

User Experience Requirements 3

System Security Requirements 3

Scalability and Maintainability Requirements 4

Work Breakdown Structure (WBS) 4

Project Initiation Phase 4

Requirements Analysis and Design Phase 5

Platform Development Phase 5

Testing Phase 5

Deployment Phase 6

Maintenance and Optimization Phase 6

Gantt chart and estimation of the total working hours for each work and total working hours and schedule 7

Overview 7

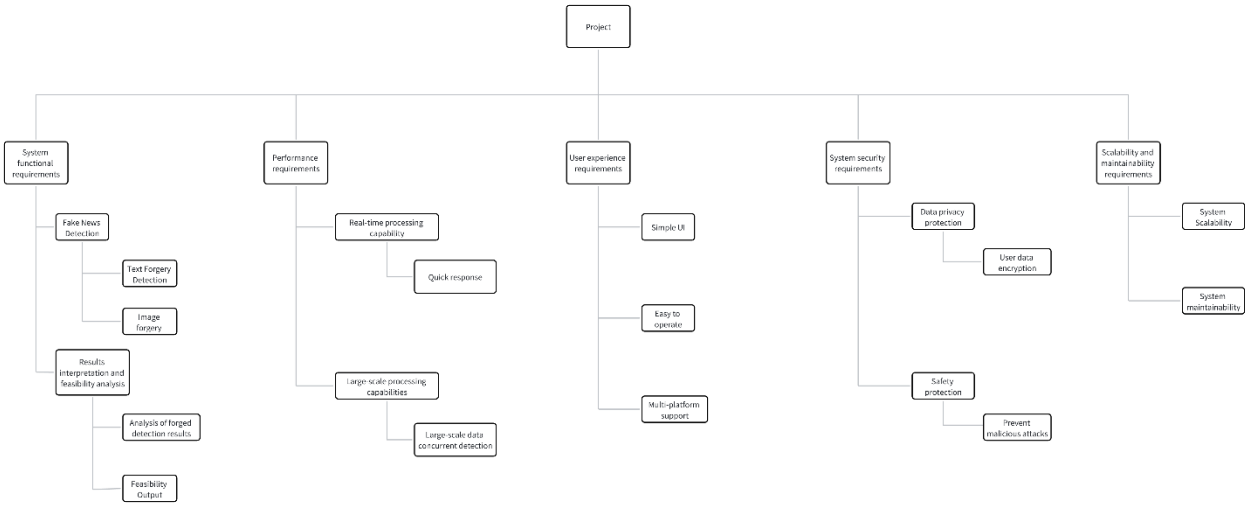
Work Breakdown and Responsibilities 7

Estimated Total Working Hours per Person 8

Total Project Workload 8

Schedule Analysis 9

RBS (requirement breakdown structure)



The clear version of RBS can be viewed through this link：

https://v1y1xrajrd9.feishu.cn/wiki/DfyGwtoZ3irlW7kTQpBc2qltnsb?from=from_copylink

Detailed explanation of RBS:

System Functional Requirements

These are the essential functions the system must be able to perform.

- **Fake News Detection**: This is the core functionality, ensuring the system can detect false news content.
    - **Text Forgery Detection**: Detects alterations or falsifications in text.
        - **Image Forgery**: Detects forged or manipulated images within the content.
- **Results Interpretation and Feasibility Analysis**: This focuses on interpreting the results from the detection system and evaluating their feasibility.
    - **Analysis of Forged Detection Results**: A task that evaluates the accuracy and reliability of the forged content detection.
    - **Feasibility Output**: Analysis regarding the practicality and scalability of the results produced.

Performance Requirements

Performance defines how well the system should perform under certain conditions, such as speed and scale.

- **Real-time Processing Capability**: The system needs to process data in real-time.
    - **Quick Response**: The system must respond rapidly to queries and actions.
- **Large-scale Processing Capabilities**: The system must handle vast amounts of data and process them efficiently.
    - **Large-scale Data Concurrent Detection**: The system must support simultaneous detection of large datasets without compromising speed.

User Experience Requirements

These are the expectations for the user interface and user experience, focusing on making the system easy to use.

- **Simple UI**: The user interface should be straightforward and easy to navigate.
- **Easy to Operate**: The system should be intuitive, requiring minimal effort for users to operate.
- **Multi-platform Support**: The system should support multiple platforms to accommodate a wide range of devices and operating systems.

System Security Requirements

This section outlines the security measures required to protect user data and the integrity of the system.

- **Data Privacy Protection**: Ensuring that user data is protected from unauthorized access or misuse.
    - **User Data Encryption**: Sensitive data should be encrypted to prevent data breaches.
- **Safety Protection**: The system must include mechanisms to safeguard against malicious threats.
    - **Prevent Malicious Attacks**: Measures to prevent attacks such as hacking or

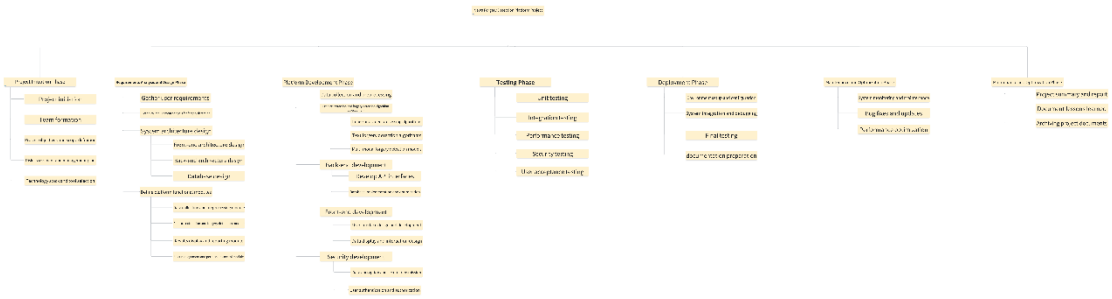
data corruption.

Scalability and Maintainability Requirements

These requirements focus on ensuring the system can grow and be maintained over time.

- **System Scalability**: The system should be able to scale efficiently as demand increases.
- **System Maintainability**: The system should be easy to maintain, with clear processes for updates and bug fixes.

Work Breakdown Structure (WBS)



The clear version of WBS can be viewed through this link：

https://v1y1xrajrd9.feishu.cn/wiki/DfyGwtoZ3irlW7kTQpBc2qltnsb?from=from_copylink

Detailed explanation of WBS:

Project Initiation Phase

- **Project initiation**: Define the project's goals and scope to ensure all team members understand the project's core objectives.
- **Team formation**: Select appropriate team members and allocate roles and responsibilities effectively.
- **Risk assessment and management plan**: Evaluate potential risks during project execution and develop mitigation strategies.
- **Technology stack and tool selection**: Choose appropriate technologies and tools

to ensure smooth development throughout the project.

Requirements Analysis and Design Phase

- **Gather user requirements**: Collect technical requirements and forgery data through communication and research with users to guide platform development.
- **System architecture design**: Design the overall system architecture, including front-end, back-end, and database design, ensuring it supports the platform's core functionalities.
- **Define platform functional modules**: Clarify the platform's functional modules, such as data collection and preprocessing, feature extraction and forgery detection, results display and reporting, and user management and access control.

Platform Development Phase

- **Data collection and preprocessing**: Develop a module for collecting and preprocessing data, ensuring the platform can efficiently process forgery data.
- **Feature extraction and forgery detection algorithm development**: Develop algorithms for detecting forgeries in images, videos, and text, using multi-modal data to enhance detection accuracy.
- **Back-end development**: Develop back-end API interfaces and optimize the database design to ensure system efficiency and stability.
- **Front-end development**: Design and implement the user interface to ensure user-friendly interaction and design.
- **Security development**: Implement data encryption and secure transmission mechanisms and design user authentication and authorization modules to protect user data and platform security.

Testing Phase

- **Unit testing**: Test each module independently to ensure its functionality is correct.
- **Integration testing**: Test the integration of all modules to ensure smooth cooperation between them.

- **Performance testing**: Test the platform's performance under heavy loads to ensure its stability and response time.
- **Security testing**: Identify and address any security vulnerabilities in the platform.
- **User acceptance testing**: Simulate real-world user scenarios to ensure the platform meets user requirements and offers a good user experience.

Deployment Phase

- **Environment setup and configuration**: Set up and configure the necessary runtime environment for the platform.
- **System integration and debugging**: Integrate the system and debug any issues that arise during the integration process, ensuring all parts work together smoothly.
- **Final testing**: Perform final functional and performance testing to ensure the platform is ready for deployment.
- **Documentation preparation**: Prepare project documentation to ensure a smooth handover for future support and maintenance.

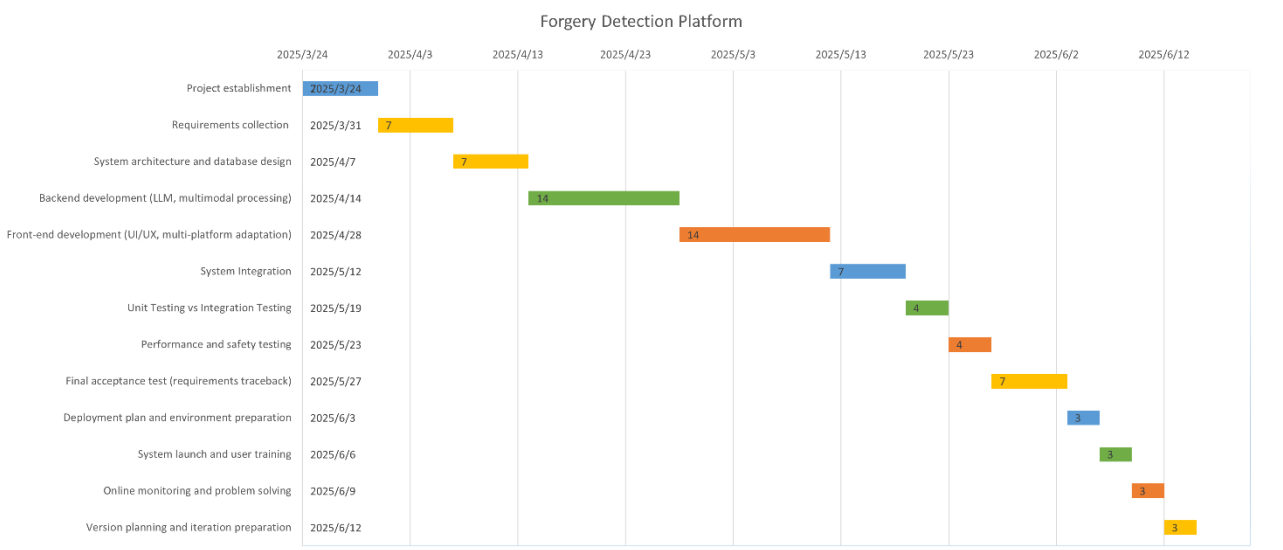
Maintenance and Optimization Phase

- **System monitoring and maintenance**: Continuously monitor the platform to ensure its proper functioning and resolve any issues promptly.
- **Bug fixes and updates**: Regularly fix bugs in the system and release updates.
- **Performance optimization**: Conduct performance tuning to keep the platform operating at its optimal state.
- **Project summary and archiving**: Prepare a project summary report, document lessons learned, and archive project documents for future reference.

Gantt chart and estimation of the total working hours for each work and total working hours and schedule

Overview

The Gantt chart illustrates the timeline and duration of key tasks involved in the **Forgery Detection Platform** development. Each task is assigned to a specific person in charge, and the duration is color-coded to differentiate responsibilities.



	Version planning and iteration preparation	Online monitoring and problem solving	System launch and user training	Deployment plan and environment preparation	Final acceptance test (requirements traceback)	Performance and safety testing	Unit Testing vs Integration Testing	System Integration	Front-end development (UI/UX, multi-platform adaptation)	Backend development (LLM, multimodal processing)	System architecture and database design	Requirements collection	Project establishment
Start Time	2025/6/12	2025/6/9	2025/6/6	2025/6/3	2025/5/27	2025/5/23	2025/5/19	2025/5/12	2025/4/28	2025/4/14	2025/4/7	2025/3/31	2025/3/24
Completion days	3	3	3	3	7	4	4	7	14	14	7	7	7

Work Breakdown and Responsibilities

Project Task	**Start Date**	**Duration (Days)**	**Person in Charge**
Project Establishment	2025/3/24	7	Zhang
Requirements Collection	2025/3/31	7	Tao
System Architecture and Database Design	2025/4/7	7	Tao
Backend Development (LLM, Multimodal	2025/4/14	14	Hu

Processing)			
Front-end Development (UI/UX, Multi-Platform Adaptation)	2025/4/28	14	Huang
System Integration	2025/5/12	7	Zhang
Unit Testing vs Integration Testing	2025/5/19	4	Hu
Performance and Safety Testing	2025/5/23	4	Huang
Final Acceptance Test (Requirements Traceback)	2025/5/27	7	Tao
Deployment Plan and Environment Preparation	2025/6/3	3	Zhang
System Launch and User Training	2025/6/6	3	Hu
Online Monitoring and Problem Solving	2025/6/9	3	Huang
Version Planning and Iteration Preparation	2025/6/12	3	Tao

Estimated Total Working Hours per Person

Assuming an **8-hour workday**, the total estimated working hours for each person can be calculated:

Person	**Total Duration (Days)**	**Estimated Hours (8h/day)**
Zhang	17	136 hours
Tao	20	160 hours
Hu	21	168 hours
Huang	21	168 hours

Total Project Workload

- **Total Duration**: 81 days
- **Total Estimated Working Hours**: 632 hours (8h/day for all tasks combined)

Schedule Analysis

- The project spans from **March 24, 2025, to June 15, 2025**.
- The **longest task** is "Backend Development (LLM, Multimodal Processing)" and "Front-end Development (UI/UX, Multi-Platform Adaptation)" at **14 days each**.
- Each phase is handled by a **specific team member**, ensuring **parallel task execution** and **efficient workload distribution**.