U-Bid Online Auction System Project Plan

SWE 6633 Section 01

Summer 2024

Team members

- Elliotte Wideman
- Connor Bland
- Hunter Blake
- Mekonnen Kindo

Table of Contents

1. Project Scope

Our U-Bid online auction platform is designed to bring sellers and buyers from all over the world through an advanced web based system and user-friendly interface. The system allows users to register, login, view or buy items on sale, list items for sale, and participate in bidding. Key features of this system include real-time auction updates, user profiles, auction records, produce reports etc. We will provide comprehensive testing and support for both desktop and mobile access to ensure a seamless user experience.

2. Functional Requirements:

2.1 User Registration and Authentication

- ☐ Users will be able to register for an account with a valid email address and create a secure login/password
- ☐ Passwords will be securely stored and transmitted

2.2 Product/Item record

- ☐ Stores an Item title and description in the database

➢ Payment Processing System

- ☐ Provides functionality regarding type of payment (check, credit-card, PayPal etc.)

2.3 GUI – User Interfaces

- ☐ Provides an interactive visual UI including:
  - **A Home page –** which user sees upon entering the site
  - **Registration page –** asks detailed information from the user and stores in the database
  - **Login page –** provides users to enter their credentials to enter into the site
  - **List Item page –** allows user to put up an item for sale
  - **Buy Item page –** allows users to buy an item from a list of items for sale

- **A payment processing page** that allows to user to input their payment information

- **A receipt page** that provides the user with a summary of the order (complete with selected items)

- **An order history page** for customers to view their past transactions

Schedule

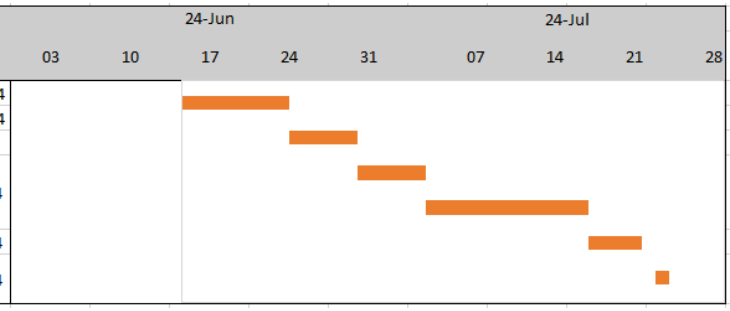
Work Breakdown Structure:

Task ID	W.B.S	Plan start	Plan finish	Workload - planned	Workload - actual	Progress (% complete)
1	**Software Project Plan Documents**	June 15	June 23	Team members will work in tandem in order to complete the following components: Gantt Chart + Scheduling, Team Organization & Roles, Data Management, Technical & Project Description, and Test Plan	Actual workload is the same as planned workload - same categories will be worked on.	100%
2	**SRS (Software Requirements Spec)**	June 24	June 28	Hunter Blake - Use-Case Diagram + Class Diagrams Connor Bland/Elliotte Wideman- Use case flow of events document Hunter Blake - Class Diagrams + Class Documentation Mekonnen/Elliotte Wideman - State-Transition Diagrams Hunter Blake - Entity-relationship diagram Mekonnen/Connor - Class Documentation	Actual workload is the same as planned workload - same categories will be worked on. Systems Design Documents may be started earlier if this phase is completed earlier as well. Programmer roles (as described here) may shift & may be worked on multiple programmers simultaneously	100% - at least 50% or more should be completed by June 25
3	**SDD (Software Design Document)**	June 29	July 3	Mekonnen - Reports Formats + class diagrams Connor Bland - Database table descriptions + class diagrams Elliotte Wideman - Technical Support Specification + class diagrams Hunter Blake - Reports formats + class diagrams	Actual workload is the same as planned workload - same categories will be worked on. Programmer roles (as described here) may shift & may be worked on multiple programmers	100% - at least 50% or more should be completed by July 1

					simultaneously	
4	**Code Review/Source code Implementation**	July 4	July 15	Mekonnen - Source code implementation Elliotte - Source code implementation Connor - Source code implementation Hunter - Source code implementation	Actual workload is the same as planned workload - same categories will be worked on.	100% - at least 50% or more should be completed by July 10
5	**User Testing + Program Testing**	July 16	July 19	Mekonnen - Development Testing Elliotte - Development Testing Connor - Development Testing Hunter - Development Testing	Actual workload is the same as planned workload - same categories will be worked on. This component of the project will be implemented during coding phases.	100% - all testing should be completed and done by July 18
6	**Final Project and Report Deliverable**	July 20	July 21			

❖ Project Schedule and Task Planning Gantt Chart

Task ID	Task Name	Duration	Start	Finish
1	Project Plan Docs	8	15-Jun-2024	23-Jun-2024
2	Requirement Docs	5	23-Jun-2024	28-Jun-2024
3	Design Docs	5	28-Jun-2024	3-Jul-2024
4	Prototypes & Implementation Docs	12	3-Jul-2024	15-Jul-2024
5	Testing Docs	4	15-Jul-2024	19-Jul-2024
6	Final Project & Report Deliverable	1	20-Jul-2024	21-Jul-2024

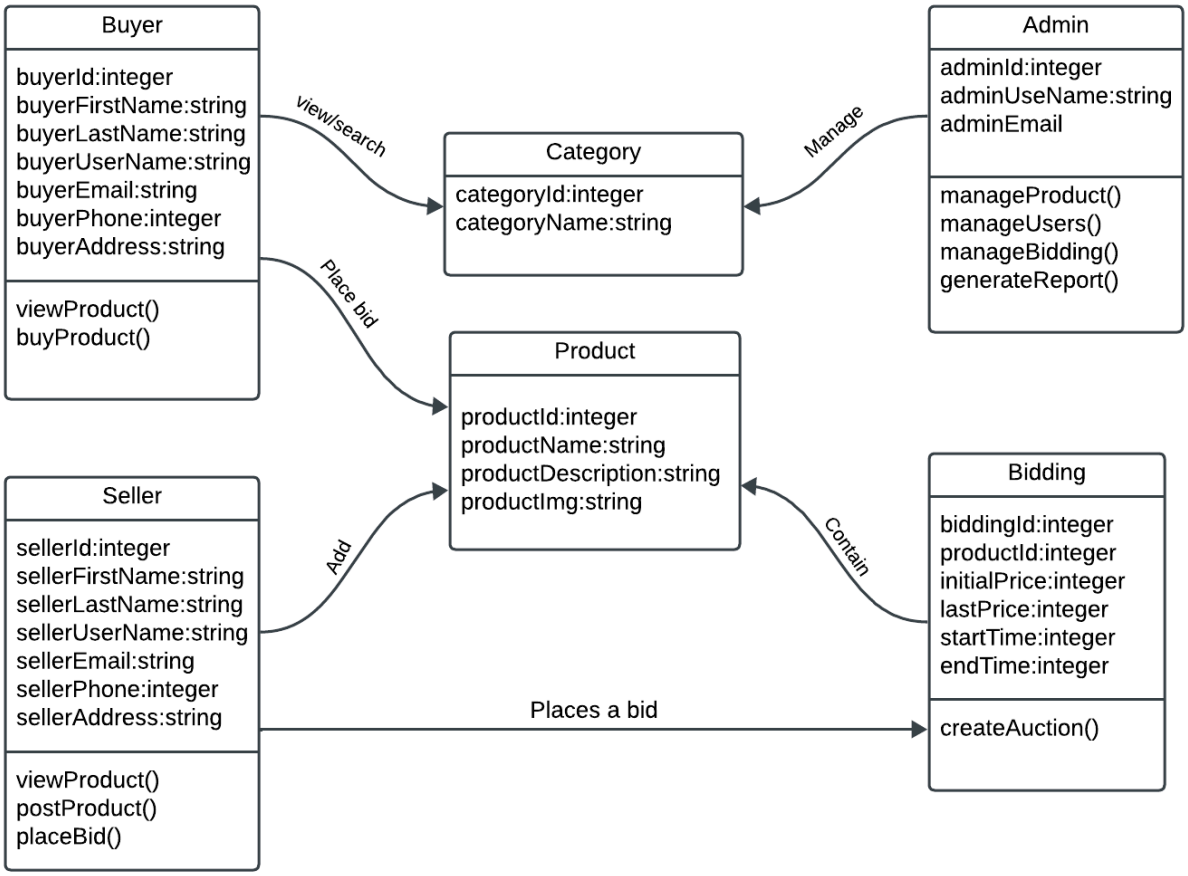


❖ Other Plans - Like Risk Assessment (if applicable)

-

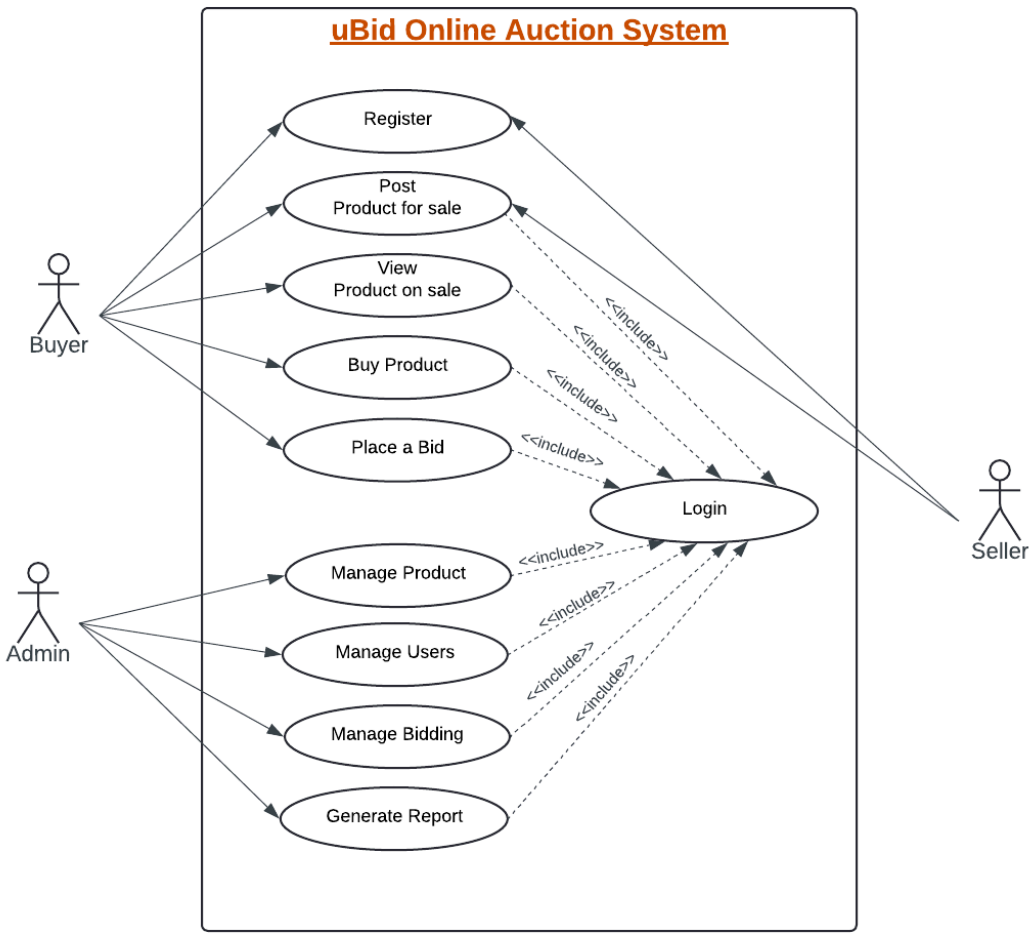
- **Version Control Plan**
  - **Git / GitHub**
  - **https://github.com/KECB24/SWE663324**

- **Class Diagram**



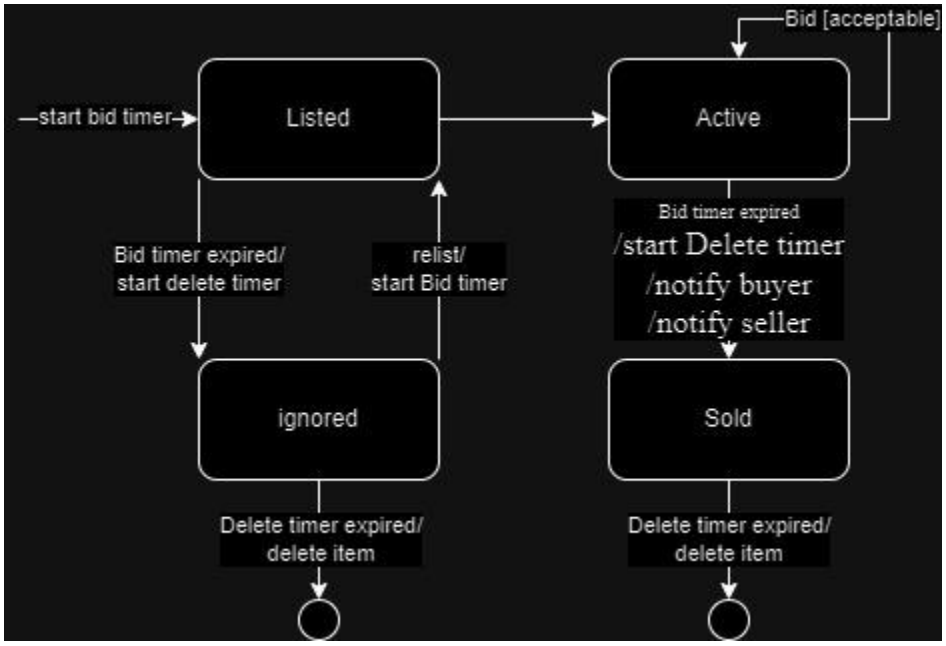
uBid online Auction system class diagram

❖ Use Case Diagram



Use Case Diagram

- State Transition Diagram



Project Overview

Project Name: UBID
Client: On-Site Auction Business
Project Duration: 6 Weeks
Team Members: Hunter Blake, Mekonnen Kindo, Connor Bland
Project Manager: Elliotte Wideman

Project Phases and Timelines

Week 1: Requirement Gathering and Analysis
- Conduct initial meetings with stakeholders.
- Gather detailed requirements through interviews and questionnaires.
- Document functional and non-functional requirements.

Week 2: Analysis and Approval
- Analyze existing on-site auction processes.
- Create a requirements specification document.
- Obtain client approval on the requirements document.

Week 3: System Design
- Design system architecture (use case diagrams, class diagrams, sequence diagrams).
- Develop database schema for users, items, categories, bids, and reports.

Week 4: UI Design and Review
- Create wireframes for user interfaces (home page, registration, login, item listing, bidding).
- Review wireframes and database schema with the client.

Week 5: Finalize Design
- Finalize system design based on client feedback.
- Prepare detailed design documents.

Week 6: Initial Implementation
- Set up development environment (servers, databases, frameworks).
- Implement user registration and login functionalities.
- Develop user profile management module.
- Begin integration and testing of implemented functionalities.

Milestones and Deliverables
- Week 2: Requirements Specification Document

- Week 5: Detailed Design Documents and Wireframes
- Week 6: Initial Implemented Modules (User Registration, Login, Profile Management)

Architectural Structure

System Architecture

Layers:
- Presentation Layer: User interfaces (UI) built with HTML/CSS.
- Application Layer: Business logic handled by PHP scripts.
- Data Layer: Data storage and retrieval managed by a relational database.

Components:
- Frontend:
  - HTML, CSS, JavaScript (for dynamic content and interactivity)
  - Bootstrap (optional, for responsive design)

- Backend:
  - PHP (using a framework like Laravel for structure, or plain PHP if preferred)
  - Apache or Nginx (web server)

- Database:
  - MySQL or PostgreSQL

- External Services:
  - Payment Gateway (Stripe, PayPal)
  - Email Service (PHPMailer, SendGrid)
  - Image Storage (local server or cloud storage like AWS S3)

Detailed Architecture
Frontend:
- Home Page:
  - Registration/Login
  - Browse Items

  - Admin Login
- Registration Page:
  - Collect user details (username, real name, password, shipping address, credit card info)
- Login Page:
  - Verify credentials
  - Redirect to user dashboard
- Dashboard:
  - Display options to buy/sell items
  - List of categories
- Item Listing Page:
  - List items for a selected category
  - Item details and bidding options

Backend:
- User Management Module:
  - Register, login, manage user profiles
- Auction Management Module:
  - Create, update, end auctions
  - Manage categories and subcategories
- Bidding Module:
  - Place and manage bids
- Reporting Module:
  - Generate reports on sales, active auctions, user activities
- Notification Module:
  - Send email notifications for bids, auction outcomes, etc.

Database Schema:
- Users Table:
  - id, username, real_name, password, shipping_address, credit_card_info
- Items Table:
  - id, title, description, category_id, user_id (seller), start_time, end_time

- Categories Table:
  - id, name, parent_id (for subcategories)
- Bids Table:
  - id, item_id, user_id (bidder), bid_amount, bid_time

Risk Management

Requirement Changes
- Mitigate by freezing requirements after initial approval.
- Use a change control process for any subsequent changes.

Technical Challenges
- Allocate time for research and prototyping.
- Regular team meetings to discuss and resolve technical issues.

Timeline Delays
- Monitor progress regularly and adjust the plan as needed.
- Maintain a buffer in the project schedule for unforeseen delays.

Communication Plan

Weekly Status Meetings
- Monday: 6:00 P.M.
- Wednesday: 6:00 P.M.
- Friday: 6:00 P.M.
- Saturday: If needed
- Update the client on progress and any issues.
- Discuss next steps and deliverables.

By following this plan, the project team can effectively develop and deploy a robust online auction system within the 6-week timeframe. Regular updates and communication will ensure alignment with client expectations and timely delivery of the final product.