**Module 2: Terminology**

Deana Akimov

Bellevue University

CSD460-O307 Capstone in Software Development (2261-DD)

Darren Osier

October 23, 2025

**Lodge**

* A type of accommodation, often in a rural or natural setting, offering lodging and recreational activities.

**Cabin**

* A small, rustic dwelling, typically made of wood, used for lodging.

**Amenities**

* Features or services offered at the lodge, such as Wi-Fi, dining, or recreational equipment.

**Occupancy**

* The number of guests currently staying at the lodge or in a specific cabin.

**Reservation**

* A booking made in advance to secure accommodation (e.g., a cabin or room) or a service (e.g., a guided tour) at the Moffat Bay lodge for a specific date and duration. This includes details such as guest name, contact information, dates of stay, and any specific requests or requirements.

**Landing Page**

* A landing page is a website page designed to achieve a specific goal, like getting visitors to sign up for a newsletter or buy a product. It's usually focused and has a clear call to action.

**Optimize**

* To improve a system or resource for the best possible result. Examples include refining the booking system for a better user experience or improving energy use to cut costs.

**API (Application Programming Interface)**

* A collection of protocols and software structure construction tools, such as a booking system to integrate with the third-party travel websites.

**Database**

* An organized collection of information stored in a computer, accessed in multiple ways. The Moffat Bay project would have a database where the information about the guests, reservations, availability of cabins, and other information would be stored.

**Frontend**

* The interface of a software program. In the case of Moffat Bay, this would be the section of the site or the application that will be used by the guests to make a reservation, see cabin details, etc.

**Backend**

* The server end of a software program, which is data storage, processing, and security. Moffat Bay would be operated by the backend that would handle the database, booking requests, and other tasks in the background.

**UI/UX (User Interface/User Experience)**

* The style and interface of a software application.

**Cloud Hosting**

* The software and data are hosted on remote servers, which can be scaled and made accessible.

**Mobile Responsiveness**

* The capability of a site or application to respond to the various sizes and types of screens.

**Bug**

* An imperfection in the code or program that makes it not do its job as might be expected.

**Code**

* The code refers to the instructions that are in a computer program, in this case, Java code.

**Deployment**

* The act of making software accessible. The implementation of updates to the Moffat Bay website would include updating the code on the web server.

**Firewall**

* A security system is one that defends against unwarranted access into a computer network. A firewall would be important in the securing of guest data and financial information of Moffat Bay.

**Algorithm**

* The cabin assignments may be optimized, using an algorithm, based on the guests' preferences and availability.

**Version Control**

* Software versioning. The version control would enable those who were working on the Moffat Bay website to easily revert to past versions in case something goes awry.

**Encryption**

* A process where the data is coded to keep it away. Sensitive data on guests, including credit card numbers, would be encrypted.

**Scalability**

* The capability of a system to process more traffic or data. The website of Moffat Bay would require scalability to accommodate peak seasons in bookings.

**Testing**

* This involves the act of testing software to make sure that it is the correct software.

**Prototyping**

* Early versions of a product are used to test ideas and designs.

**Personas**

* These are fictional characters created to represent the different user types that might use a site, brand, or product in a similar way.

**User Stories**

* Simple descriptions of a feature are told from the perspective of the person who desires the new capability, usually a user or customer of the system. The user story describes who wants the feature, what they want to accomplish, and why.

**Story Cards**

* These are physical or digital cards used in Agile development to represent user stories. They typically include a brief description of the feature, an estimate of the effort required to implement it, and acceptance criteria.

**Planning Poker**

* Consensus-based techniques are used to estimate the effort required to implement user stories or other tasks.

**Work Estimation**

* Predicting the effort, time, and resources needed for a task.

**Kanban**

* A visual system for managing workflow, improving efficiency by visualizing tasks, and limiting work in progress.

**UI Design**

* Creating the visual and interactive elements of a product for an intuitive user experience.

**Entity Relationship Diagram**

* A visual way to show how different data elements (entities) relate to each other in a system.

**SQL Statement**

* A command to interact with a database for tasks like retrieving or modifying data.

**Sprint**

* A short, time-boxed period when a scrum team works to complete a set amount of work.